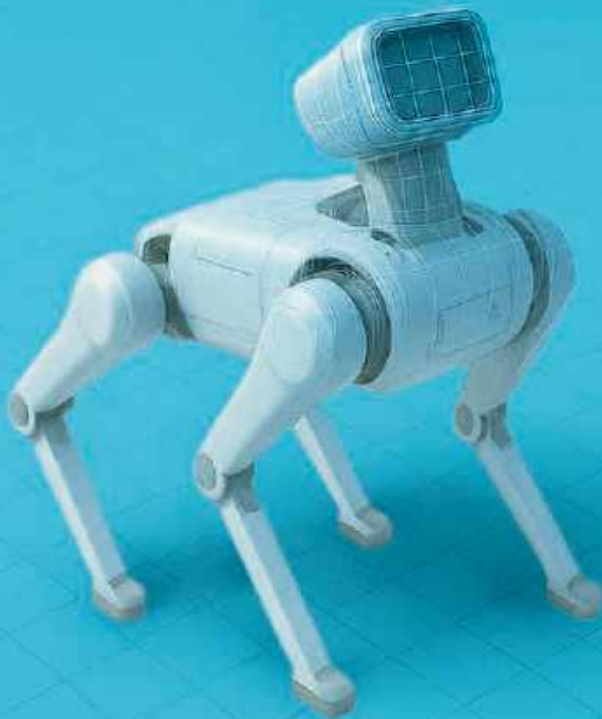




Volume 2

Readings in AI

2024/25 Publications of the ZHAW
Centre for Artificial Intelligence



Imprint

Prof. Dr. Thilo Stadelmann
CAI Centre for Artificial Intelligence
ZHAW Zurich University of Applied Sciences
Technikumstrasse 71
8401 Winterthur
office.cai@zhaw.ch

All publications are reprinted from their publicly available sources (open access, pre-print) and used with permission. Copyrights belong to their respective holders.

Cover: Nerves GmbH, Zürich

Preface

The ZHAW Centre for Artificial Intelligence (CAI) is a hub for excellence in AI research and application. Our mission is to advance human-centric and trustworthy AI research in Switzerland and beyond, thereby providing students with career opportunities, attracting young talent and addressing the great challenges of our time through innovative use of AI.

We believe in the power of interdisciplinary collaboration and engaging in dialogue with the research community, with our students, and with our partners outside academia. We offer expertise in the areas of *Autonomous Learning Systems* (reinforcement learning, multi-agent systems, and embodied AI), *Computer Vision, Perception and Cognition* (pattern recognition, machine perception, and neuromorphic engineering), *Natural Language Processing* (dialogue systems, text analytics, and spoken language technologies), *Trustworthy AI* (explainable AI, robust deep learning, and AI & society) and *AI Engineering* (MLOps, data-centric AI, and continuous learning). The CAI's vision is to contribute to a society that is worth living in, with humans at the centre.

As the hype and adoption of generative AI accelerated throughout 2024 and into Q1/2025 (the period this report is based on), CAI members became even more busy with public statements, talks, interviews and educational offers geared towards the general audiences. One result of this will be a new CAS on AI essentials without any programming requirements. We expect to see our involvement in continuing education grow. At the same time, we witnessed a shift in applied research demand: Many new projects are now built on available foundation models, while challenges that kept us busy for years (like certain classification tasks on images, texts, etc.) appear less needed. A sign that technology is maturing, and the frontier of research is moving on.



The CAI continued to grow in 2024, with Alisa Rupenyan's Industrial AI group increasing the most and the NLP group being the largest again. Ricardo Chavarriaga, together with associated faculty member Christoph Heitz, was able to acquire the next DIZH structure project, AI-BRIDGE, that will keep more than his Responsible AI Innovation group busy in coming years. We also welcomed Manuel Renold as new group leader for Cybernetic Learning Systems. He invested himself heavily and very successfully in project acquisition, lab buildup, and teaching. By January 2025, he and his team moved on to the FHNW School of Business. We will replace the lectureship on physical AI and RL soon. Meanwhile, the first fruits of our collaboration with the Swiss AI Initiative became visible, and we opened two lab spaces on robot learning and mobile robotics.

With this research report, we issue the fourth account of our work, now including also the first quarter of the current year. From news, to teaching, to the individual groups' research output – enjoy reading volume 2!

Winterthur, Spring 2025



Prof. Dr. Thilo Stadelmann, Director of Centre for Artificial Intelligence

Table of Contents

Preface	I
Table of Contents	II
7 Research Output of the Machine Perception and Cognition Group	1
Research Output of the Machine Perception and Cognition Group	3
A Comprehensive Survey of Deep Transfer Learning for Anomaly Detection in Industrial Time Series: Methods, Applications, and Directions	5
Real World Music Object Recognition.....	27
Automated process monitoring in injection molding via representation learning and setpoint regression	41
So you want your private LLM at home? A survey and benchmark of methods for efficient GPTs.....	49
MathNet: A Data-Centric Approach for Printed Mathematical Expression Recognition	57
ScalaGrad: A Statically Typed Automatic Differentiation Library for Safer Data Science	69
Deep Neural Networks for Automatic Speaker Recognition Do Not Learn Supra-Segmental Temporal Features	73
Machine Learning for Robust Structural Uncertainty Quantification in Fractured Reservoirs	81
Deep-learning-based Cell Segmentation for Rapid Optical Cytopathology of Thyroid Cancer.....	103
Community-Hop: Enhancing Node Classification through Community Preference	125
Hierarchical Glocal Attention Pooling for Graph Classification	135
A Simulation Study on Energy Optimization in Building Control with Reinforcement Learning	143
A guide to AI: Understanding the technology, applying it successfully, and shaping a positive future	155
8 Research Output of the Natural Language Processing Group	173
Research Output of the Natural Language Processing Group.....	175
Improving Quantification with Minimal In-Domain Annotations: Beyond Classify and Count.....	177
Error-preserving Automatic Speech Recognition of Young English Learners' Language	191
Favi-Score: A Measure for Favoritism in Automated Preference Ratings for Generative AI Evaluation.....	203
Role-Playing LLMs in Professional Communication Training: The Case of Investigative Interviews with Children	221
9 Research Output of the Responsible AI Innovation Group	237
Research Output of the Responsible AI Innovation Group	239
Customizing the human-avatar mapping based on EEG error related potentials	241

MLOps as Enabler of Trustworthy AI	255
Towards the Certification of AI-based Systems	259
A Certification Scheme for Artificial Intelligence based Systems.....	267
SSBE 2024: Python-based Open Software for EEG-based Brain-Machine Interfaces	277
Evaluating dry EEG technology out of the lab.....	279
certAInty – A Certification Scheme for AI Trustworthiness An Innosuisse Project.....	285
AI Assessment in Practice: Implementing a Certification Scheme for AI Trustworthiness	287
Finding Consensus on Trust in AI in Health Care: Recommendations From a Panel of International Experts.....	305
The NERVE-ML (Neural Engineering Reproducibility and Validity Essentials for Machine Learning) Checklist: Ensuring Machine Learning Advances Neural Engineering	323
Appendix	371
A.1 List of Publications – Volume 2	371
A.2 CAI Team as of 31.03.2025.....	375
A.3 Location	377

7 Research Output of the Machine Perception and Cognition Group

7 Research Output of the Machine Perception and Cognition Group

The MPC group, led by Prof. Stadelmann, performs project-based research in representation learning for pattern recognition, working on a wide variety of tasks on image, audio or generally “signal” data.

As a group, we focus on deep learning methodology, inspired by biological learning. Each task that we study has its own learning target (e.g., detection, classification, clustering, segmentation, novelty detection, control) and use case (e.g., predictive maintenance, speaker diarization for multimedia indexing, document analysis, optical music recognition, computer vision for industrial quality control, automated machine learning, deep reinforcement learning for automated game play or building control), which in turn sheds light on different aspects of the learning process. We also engage in interdisciplinary work on the ethical and societal aspects of AI.

During the period 2024/25, Dr. Lukas Tuggener and Dr. Waqar Ali successfully defended their PhD theses; Lukas went on full-time to his start-up RWAI. Raphael Emberger and Erdal Ayfer left us in the Fall, while Dr. Hella Bolck (as senior scientific project manager) and Waqar Ali (to finish his PhD and PostDoc) came on board. Prof. Dr. Stadelmann spent the Fall term 2024 on sabbatical leave, talking about the group's work and future collaborations in Switzerland (UZH/ETH), the UK (Universities of Cambridge and Oxford), the USA (New York University, Columbia, WPI, and UMass Lowell), "down under" (QUT and Victoria University of Wellington) and in Germany (FAU Erlangen-Nürnberg).

While several long-standing projects are ending soon (DISTRAL on industrial process control and Master3D on document intelligence), the group successfully acquired new projects (e.g. dAIrector on automated multi-camera video direction, and EchoRapp-AI on diagnostic assistance for echocardiography) and collaborations (for example with cantonal institutions on strategic foresight, and the Mindfire Foundation on physical AI). Our engagement in autonomous systems (e.g., project LINA) and societal effects of AI (e.g., human- AI colearning in project AI4REALNET) continues.

Publication-wise, there was a rich harvest also from papers submitted in previous years. We saw a huge increase in journal publications (among others, 2x IEEE Access, Transaction of ISMIR, 2x Pattern Recognition Letters, Nature Scientific Reports), the final publication of some longer-standing collaborations (on geothermics, RL for building control, and cancer diagnostics), and a successful participation at the 11th Swiss Conference on Data Science with 3 papers (one an honorable mention to the best paper award: “So you want your private LLM at home?”). Four ArXiv uploads in Q1/2025 speak of more papers to come out soon.

We thank our project partners, students, and funding bodies for their generous support and effort, without which, these results (and future results) would not have been possible!

The MPC group, as of March 2025:



Waqar Ali, Pascal Sager, Thilo Stadelmann, Ahmed Abdulkadir, Benjamin Meyer, Peng Yan, Hella Bolck (with 5 MSc students, 1 external PhD student and 3 associated faculty members)

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2023.0322000

A Comprehensive Survey of Deep Transfer Learning for Anomaly Detection in Industrial Time Series: Methods, Applications, and Directions

PENG YAN^{1,2}, AHMED ABDULKADIR¹, PAUL-PHILIPP LULEY¹, MATTHIAS ROSENTHAL³, GERRIT A. SCHATTE⁴, BENJAMIN F. GREWE², and THILO STADELMANN^{1,5}

¹Centre for Artificial Intelligence, ZHAW School of Engineering, Winterthur, Zürich 8400 Switzerland²Institute of Neuroinformatics, ETH & University of Zürich, Zürich, Zürich 8057 Switzerland³Institute of Embedded Systems, ZHAW School of Engineering, Winterthur, Zürich 8401 Switzerland⁴Innovation Lab, Kistler Instrumente AG, Winterthur, Zürich 8408 Switzerland⁵ECLT European Centre for Living Technology, Venice, Veneto 30123 Italy

Corresponding author: Peng Yan (e-mail: yanp@zhaw.ch).

This work has been supported by Innosuisse under Grant 62174.1 IP-ENG “DISTRAL”.

ABSTRACT Automating the monitoring of industrial processes has the potential to enhance efficiency and optimize quality by promptly detecting abnormal events and thus facilitating timely interventions. Deep learning, with its capacity to discern non-trivial patterns within large datasets, plays a pivotal role in this process. Standard deep learning methods are suitable to solve a specific task given a specific type of data. During training, deep learning demands large volumes of labeled data. However, due to the dynamic nature of the industrial processes and environment, it is impractical to acquire large-scale labeled data for standard deep learning training for every slightly different case anew. Deep transfer learning offers a solution to this problem. By leveraging knowledge from related tasks and accounting for variations in data distributions, the transfer learning framework solves new tasks with little or even no additional labeled data. The approach bypasses the need to retrain a model from scratch for every new setup and dramatically reduces the labeled data requirement. This survey first provides an in-depth review of deep transfer learning, examining the problem settings of transfer learning and classifying the prevailing deep transfer learning methods. Moreover, we delve into applications of deep transfer learning in the context of a broad spectrum of time series anomaly detection tasks prevalent in primary industrial domains, e.g., manufacturing process monitoring, predictive maintenance, energy management, and infrastructure facility monitoring. We discuss the challenges and limitations of deep transfer learning in industrial contexts and conclude the survey with practical directions and actionable suggestions to address the need to leverage diverse time series data for anomaly detection in an increasingly dynamic production environment.

INDEX TERMS Deep transfer learning, time series analysis, anomaly detection, manufacturing process monitoring, predictive maintenance

I. INTRODUCTION

A. MOTIVATION AND CONTRIBUTION

The fourth industrial revolution – Industry 4.0 [1], that is characterized by increasing efficiency through the digitization of production, automation, and horizontal integration across companies [2], and the advent of connected cyber-physical systems – referred to as internet of things [3]–[5], increases the need for autonomous and intelligent process

monitoring. This can be exemplified by the use case of a smart factory in which industrial processes are transformed to be more flexible, intelligent, and dynamic [6], or the use case of decentralized energy production with wind and solar [7]. In these examples, AI-powered anomaly detection integrates the analysis of time series data to detect unusual patterns in the recorded data. To achieve this, a deep learning architecture is modeled to capture indicators of normal and abnormal operation. The learning process involves the analysis of historic

time series sensor data of normal and possibly abnormal operations. This data is for example used for *representation- or reconstruction-based* learning. After training, the deep learning model *represents* or *reconstructs* normal data in a certain way. The model is designed in a way that abnormal data—because it is different—is either *represented* differently from the normal data or *reconstructed* poorly and thus recognized as an anomaly. By identifying operational parameters that fall outside a window of normal interval, operators can trigger interventions and adjustments to ensure high product quality and safe operations. To achieve this, physical properties such as pressure or temperature are monitored and analyzed in real-time applications. Changes in these variables capture drifting and abrupt faults caused by process failures or malfunctions [8]. The production process must adapt quickly to changes in production and the environment to meet the requirements for flexibility and dynamics. Further use cases exist in a wide range of diverse fields, such as manufacturing monitoring including automatic quality control [9], [10], predictive maintenance of goods and services [11]–[16], infrastructure monitoring of building systems [17], [18] and power plant [19], digital agriculture [20], petrochemical process optimization [21], computer network intrusion detection [22], or aircraft flight monitoring [23], to name a few.

Artificial Intelligence, particularly deep learning, provides competent frameworks with underlying deep neural networks to automate intelligent monitoring and provide valuable assistance to operators and high-level control systems. Leveraging the power of deep learning, informative features of the data – technically referred to as *representations* [24] – can be captured in a machine-learned model and thereby enable a detailed understanding of variations in standard operations.

However, the task or underlying data may change under non-trivial and non-stationary conditions. For instance, the monitoring system of a milling machine may be assigned the task of identifying a blunt tool based on vibration in one scenario, and in a different scenario, it may utilize the same vibration measurements to detect insufficient cooling lubricant. Knowledge acquired to solve one task in one setting with a given tool, machined part, and type of machine may be transferred to solve the same or similar task in another setting with a different tool, machined part, or type of machine. Slowly changing conditions (drifts), abrupt mode changes (for instance, due to tool change), and new tasks (such as the detection of another failure mode) may require adjustments to the deep learning model. In these cases, it is desirable to adjust the analysis model without retraining from scratch, as it is costly or impractical to acquire sufficient training data to learn the full manifold [25].

Transfer learning is a machine learning framework to achieve this [26]–[30]. As depicted in Fig. 1, data and algorithms from one task may be leveraged in a new related one. By accounting for changes in data distributions and tasks or leveraging existing models, knowledge learned from related tasks can be used to improve performance on new tasks instead of retraining a model for each individual appli-

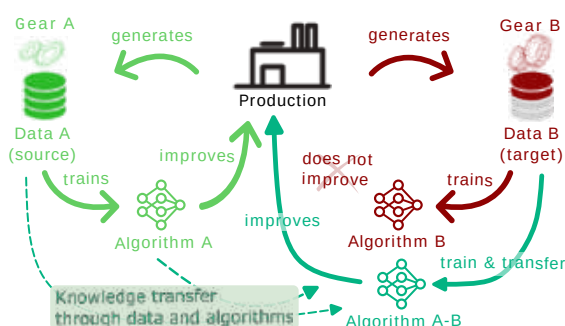


FIGURE 1. Transfer learning is useful when changes in production take place and sufficient data for full retraining is not available as shown here for a hypothetical production of two types of gears. In the production of gear A, a lot of data is available to train a deep learning model that helps improve production. In the production of gear B, data is more limited, and the traditionally trained deep learning model fails to improve production. With suitable transfer learning methods, however, data and algorithms acquired during the production of gear A can be leveraged to support improving the production of gear B because the data and tasks in the production of both gears are related.

cation from scratch. This transfer-learning-boosted modeling forms the basis for identifying anomalies that deviate from established patterns in a non-trivial manner without full re-training.

Deep transfer learning [29], [31] extends the transfer learning paradigm by leveraging deep learning. In industrial contexts, it ensures optimal production even as production conditions shift. This dynamic adaptability is key in maintaining the effectiveness of anomaly detection systems in the dynamic environment that characterizes industrial applications including the broad categories of manufacturing process monitoring, predictive maintenance, energy management, and infrastructure facility monitoring as detailed in Section IV.

This survey is a non-systematic yet application-oriented review with a narrow focus on deep transfer learning for anomaly detection in time series in the industry. Our main contributions are as follows:

- We categorize transfer learning problem settings and then systematically summarize deep transfer learning approaches into four categories. With the foundations of deep transfer learning, we equip the reader with a working knowledge of the main principles and intuitions.
- We analyze the recent literature and provide a comprehensive overview of the current state of the art of deep transfer learning approaches for time series anomaly detection for main industrial applications.
- We discuss potential challenges and limitations and then give directions for future work with actionable recommendations for AI practitioners and decision-makers.

To our knowledge, this is the first survey of deep transfer learning in the narrow context of industrial time series anomaly detection. The review describes the underlying methodological principles and methods within a generic

taxonomy and discusses practical implications for AI practitioners to make informed decisions. We cover multiple areas of application, including manufacturing monitoring, maintenance prediction, and infrastructure monitoring.

The rest of the paper is organized as follows. First, we provide an overview of transfer learning by introducing a taxonomy of transfer learning problem settings and further categorizing deep transfer learning approaches (Section II). Then, we describe the task of anomaly detection in time series (Section III) in selected industrial applications (Section IV). To conclude, we discuss current challenges, limitations, and future research directions (Sections V–VI) in the field.

B. SURVEY METHODOLOGY

We seek to identify application-oriented peer-reviewed literature in the intersection of transfer learning as the learning framework, time series as the data domain, and anomaly detection as the task (Fig. 2). To execute the selection process of literature, we search related terms on Google Scholar, Scopus, Elsevier, and IEEE databases. Based on the title, we pick those papers that may fit the narrow topic into a pre-selection list. Eventually, we included publications matching the topic according to the abstract and screening of the content.

Along the reviewed topical papers, we include contextually relevant papers such as deep learning approaches that are agnostic to data types and tasks. For deep transfer learning in general, we searched the keywords “transfer learning” and “deep transfer learning”. Specifically, we focus more on deep transfer learning approaches. Then, we switch to the application-oriented cases where deep transfer learning is applied to tackle time series anomaly detection in the main industrial applications. To achieve this, we search queries like “deep transfer learning for time series anomaly detection” and “deep transfer learning for predictive maintenance”. After searching in the database, we carefully check and screen out the most relevant literature based on the following inclusion/exclusion criteria: (1) We only include the applications that utilize deep transfer learning approaches, instead of traditional transfer learning; (2) We only include publications after 2013; (3) We cover all three main topics in Fig. 2 (highlighted with cycles), but we specifically focus on the intersection of the three aforementioned domains. After carefully screening out, we select 45 papers for deep transfer learning in general and 37 papers for deep transfer learning for anomaly detection in industrial time series.

Fig. 3 illustrates the taxonomy in this survey to categorize reviewed studies based on different aspects, including deep transfer learning, time series anomaly detection, industrial applications, current challenges, and future directions.

II. DEEP TRANSFER LEARNING

A. OVERVIEW OF THE FIELD

Transfer learning in the setting of industrial time series analysis for anomaly detection is a tool to increase the flexibility of autonomous process monitoring. It addresses the challenge of adapting the algorithm, and thus the decision process, to a

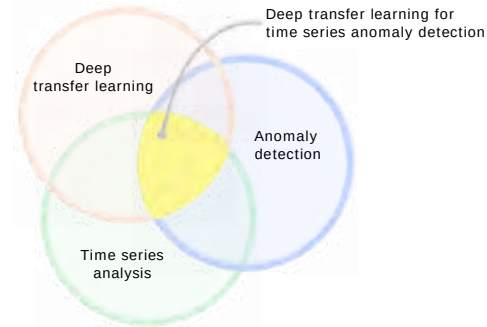


FIGURE 2. Venn diagram of this survey’s focus on the intersection of transfer learning, anomaly detection, and time series analysis.

related but previously unseen setting where limited training data is available. The transfer eliminates the need to train a deep learning model from scratch, which in turn reduces the amount of necessary data and compute required to solve a new task or adjust to a new data domain. In either case, knowledge is transferred from a source to a target domain, as described below. The transfer learning problem settings can be categorized as inductive or transductive transfer depending on the data and task conditions. We categorize deep learning-based transfer learning approaches into instance transfer, parameter transfer, mapping transfer and domain-adversarial transfer. We illustrate them by using two intuitive examples in Fig. 4, with more details being elaborated in the following sections.

B. FORMAL DESCRIPTION OF DEEP TRANSFER LEARNING

Domain $\mathcal{D}$ includes the domain feature space $\mathcal{X}$ and marginal data distribution $P(X)$ as $\mathcal{D} = \{\mathcal{X}, P(X)\}$, where X is the domain data, $X = \{x_1, \dots, x_n\} \in \mathcal{X}$. Similarly, a learning task is defined as $\mathcal{T} = \{\mathcal{Y}, f_{\mathcal{T}}(\cdot)\}$, where $\mathcal{Y}$ denotes the task space and usually represents class label. For anomaly detection tasks, $\mathcal{Y}$ is the set of the two classes “normal” and “abnormal”. The function $f_{\mathcal{T}}(\cdot)$ can be used to predict the corresponding label of a new instance x_i . The objective predictive function $f_{\mathcal{T}}(\cdot)$ learned from domain data can be interpreted as a form of conditional probability. Thus, the learning task can be rewritten as $\mathcal{T} = \{\mathcal{Y}, P(Y|X)\}$, where $P(Y|X)$ is used as a likelihood measure to determine how well a given data set X fits with a corresponding class label set Y .

In the surveyed literature on transfer learning for anomaly detection in industrial applications, transfer learning methods from other fields, such as computer vision and natural language processing, were adopted. We therefore use a generic classification scheme for transfer learning methods. We largely follow the definition of transfer learning in literature [27], [28]. Given a source domain $\mathcal{D}_S$ and learning task $\mathcal{T}_S$, as well as a target domain $\mathcal{D}_T$ and learning task $\mathcal{T}_T$, transfer learning aims to improve the performance of the

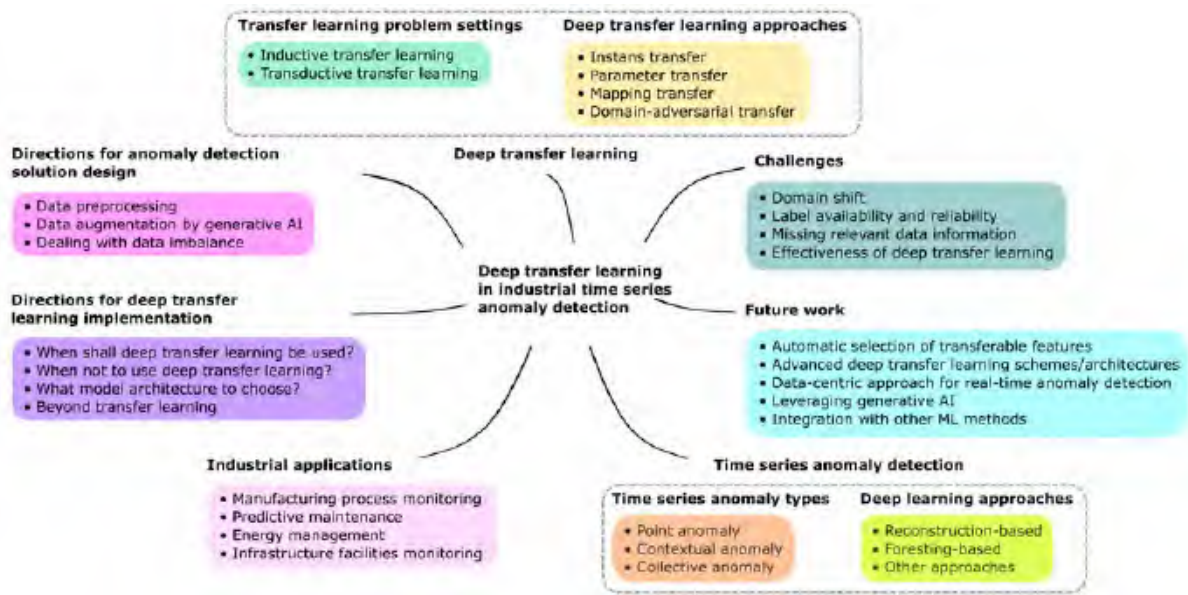


FIGURE 3. A generic taxonomy in this paper to analyze deep transfer learning for industrial time series anomaly detection.

predictive function $f_T(\cdot)$ in $\mathcal{D}_T$ by transferring knowledge from $\mathcal{D}_S$ and $\mathcal{T}_S$, where $\mathcal{D}_S \neq \mathcal{D}_T$ and/or $\mathcal{T}_S \neq \mathcal{T}_T$. Usually, the size of source dataset is much smaller than target dataset.

This definition of transfer learning can be broadened, i.e., the target task can benefit from multiple source domains. Transfer learning is thus the idea of making the best use of related source domains to solve new tasks. In contrast, traditional machine learning (ML) methods learn each task separately from scratch, and each respective model can only be applied to the corresponding task.

We define a taxonomy of transfer learning problem settings as shown in Fig. 4 mainly depending on the label availability in the two domains to be easily applicable to the requirements of a case at hand (compare different definitions for other purposes in the literature [27] – [29]).

We differentiate it into inductive and transductive transfer learning [28]. Inductive transfer learning is applied when the target task is different from the source task, i.e., $\mathcal{T}_S \neq \mathcal{T}_T$ (meaning that $\{\mathcal{Y}_S \neq \mathcal{Y}_T\}$ or $\{P(Y_S|X_S) \neq P(Y_T|X_T)\}$). The conditional probability distribution is induced with labeled training data in the target domain [34]. A corresponding example is illustrated as Scenario A in Fig. 4, where the learning tasks are different and the goal of transfer learning is to recognize point anomaly from the collective anomaly task. Related areas of inductive transfer learning are multi-task learning [35], [36] and sequential learning, depending on whether tasks are learned simultaneously or sequentially.

Transductive transfer learning is applied when the source and target tasks are the same, while the source and target domain are different, i.e., $\mathcal{T}_S = \mathcal{T}_T$ and $\mathcal{D}_S \neq \mathcal{D}_T$ (meaning that

$\{\mathcal{X}_S \neq \mathcal{X}_T\}$ or $\{P(X_S) \neq P(X_T)\}$). A subcategory is domain adaptation [37] when the feature space of source and target data are the same but the corresponding marginal distributions are different (i.e., $\{\mathcal{X}_S = \mathcal{X}_T\}$ and $\{P(X_S) \neq P(X_T)\}$). Scenario B in Fig. 4 is an example of transductive transfer learning where the learning tasks are identical, and the goal of transfer learning is to recognize contextual anomalies in an unlabelled data set.

C. DEEP TRANSFER LEARNING APPROACHES

Since deep neural networks (DNNs) can learn useful feature representations from large amounts of data through back-propagation [24], they have been widely adopted for tackling complex problems in practice [38]–[41], which involve large-scale and high-dimensional data. Deep transfer learning methods implement transfer learning principles within DNN and, among other things, enable deep learning based analysis pipelines to be applied to new datasets.

Based on the transferring techniques in the surveyed literature, we access how knowledge is shared across domains and help increase the performance in the target task or domain. We divide deep transfer learning approaches further into 4 categories: instance transfer, parameter transfer, mapping transfer, and domain-adversarial transfer, as illustrated in Table 1. Furthermore, instance transfer, mapping transfer, and domain-adversarial transfer can be described as data-driven approaches. They focus on transferring knowledge by leveraging a large amount of data. It usually involves transforming and adjusting the data instances or manipulating data from different domains by feature alignment, feature mapping,

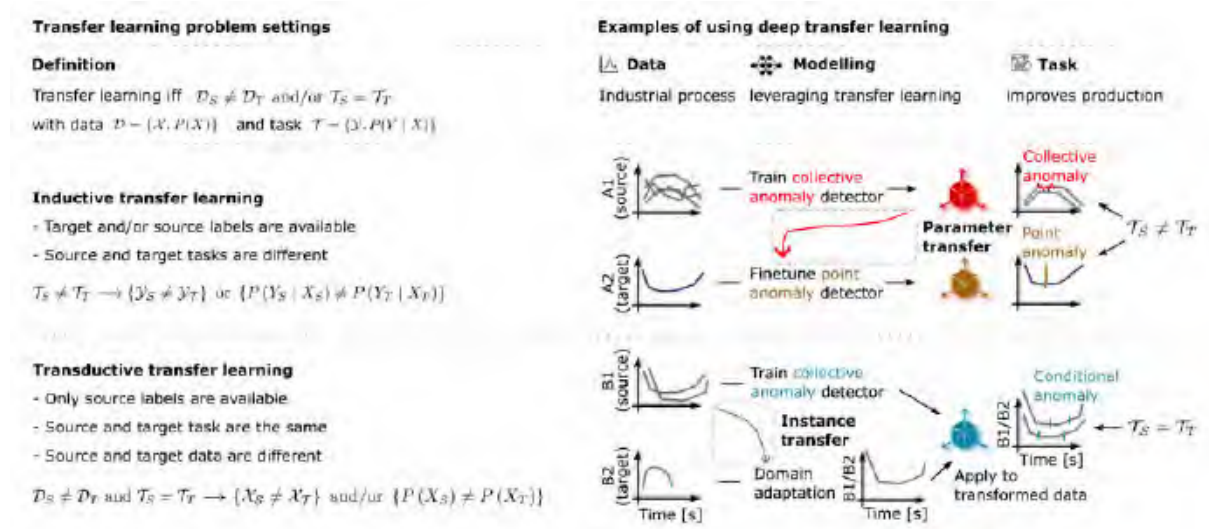


FIGURE 4. Taxonomy of transfer learning problem settings (left; see Section II-B for the definition of terms) and corresponding examples using deep transfer learning approaches (right). On the left, we classify transfer learning problems as inductive or transductive transfer settings. Correspondingly, we provide two examples using deep transfer learning methods: In the inductive transfer setting, we collect time series data from screw production and wrench production. Labeled screw data (A1) is used to detect collective anomalies (a set of data points behaving differently compared to the entire time series [32], [33], further explained in Section III). Then, parameter transfer (Section II-C2) is applied to transfer knowledge by fine-tuning the pre-trained model from labeled screw data to detect point anomalies (further explained in Section III) on labeled wrench data (A2). For the transductive transfer setting in the lower panel, we present a different situation for contextual anomaly detection (further explained in Section III). In this case, we have two datasets, B1 and B2, analyzed using the same model. However, the data in B2 significantly differs in appearance from the data in B1. To address this problem, instance transfer (further explained in Section II-C1) is used. Through this learning process, the data in B2 is transformed in a way that makes it compatible with the model that has been trained exclusively on data from B1. Transfer learning, in this case, is thus achieved by adapting the data to fit the model through domain adaptation rather than adjusting the model to fit new data.

etc. On the other hand, parameter transfer is a model-driven approach, which places more emphasis on understanding the underlying structure and dynamics of the data. It usually involves transferring the parameters of pre-trained model from source domain to target domain.

1) Instance transfer

The intuition of instance transfer is that although source and target domains differ, it is still possible to transform and reuse source data together with a few labeled target samples. A typical approach is to re-create some labeled data from the source domain. For example, He *et al.* propose an instance-based deep transfer learning model with an attention mechanism to predict stock movement [42]. They first create new samples from the source dataset that are similar to the target samples by using an attention network and then train another network on the created samples and target training samples for prediction tasks. Since two networks are trained separately for different tasks, it needs further investigation to what extent the generated samples can contribute to the prediction task. Amiraïn *et al.* introduce an innovative instance transfer method for domain adaptation [43]. They propose an effective auto-encoder model with a pseudo-label classifier to reconstruct new data instances that obtain general features across different datasets for medical image analysis. Taking another avenue, Wang *et al.* exclude the source data that negatively

impacts training target data. Specifically, they choose a pre-trained model from a source domain, estimate the impact of all training samples in the target domain, and remove samples that lower the model's performance. Then, the optimized training data is used for fine-tuning. The experiments are conducted on large image datasets [44]. Instead of transferring the data, the approach excludes certain samples based on the pre-trained model's predictions. Additional validation is required in industrial environments, especially when only a few data are available in some industrial settings.

2) Parameter transfer

Parameter transfer adapts the learned parameters of a pre-trained model to a new model. This assumes that DNNs can get similar feature representations from similar domains. Thus, through transferring parts of the DNN layers together with pre-trained parameters and/or hyperparameters, the pre-trained model is used as a base model to further train on target domain data and solve different learning tasks. Particularly, parameter transfer has gained popularity in computer vision and natural language processing, where large models are pre-trained on large datasets [45]. In natural language processing, for example, BERT [46] and GPT-3 [53] are based on the Transformer architecture [48] which can be fine-tuned for a variety of downstream tasks, including content generation [49], language translation [66], question answering [67], and

TABLE 1. Overview of deep transfer learning approaches with references.

Deep transfer learning approach	Short description	References
Instance transfer	Augmenting target data by transforming data instance from the source domain to the target domain	[42]–[44]
Parameter transfer	Transfer learned parameters of a pre-trained model from source domain and adapting the model for target domain	[17], [45]–[53]
Mapping transfer	Reducing feature discrepancies between source and target domains by minimizing the distance between mapped features in the latent space	[27], [54]–[59]
Domain-adversarial transfer	Extracting an indiscriminate feature representation between source and target domain through adversarial training	[60]–[65]

summarization [50]. In computer vision, Yosinski *et al.* investigate the general transferability of CNNs in image recognition [30]. They analyze the transferring effect by fine-tuning or freezing a certain amount of layers in the networks. Experimental results show that transferring features from source to target domain improves network generalization compared to those trained solely on the target dataset. Additionally, they quantify the model performance by assessing how features at what layers transfer from one task to another. It is surprising to find that transferring a pre-trained network from any number of layers can produce a boost for fine-tuning on a new dataset. However, the experiments are only conducted on certain image datasets, and Tugeneer *et al.* show [68] the limits of parameter transfer when the chosen architecture is overfitted on the particularities of certain large-scale datasets.

Unlike the typical way of fine-tuning a pre-trained model, Guo *et al.* propose an adaptive fine-tuning approach SpotTune to find the optimal fine-tuning strategy for the target task [51]. Specifically, a policy network is used to make routing decisions on whether to pass the target instance through the pre-trained model. The results show SpotTune is effective in most cases by using a hybrid of parameter and instance transfer. Sager *et al.* propose an unsupervised domain adaptation for vertebrae detection in 3D CT volumes by transferring knowledge across domains during the training process [52].

3) Mapping transfer

Mapping transfer refers to learning a related feature representation for the target domain by feature transformation, which includes feature alignment, feature mapping, and feature encoding [27]. The goal is to reduce feature discrepancies between source and target domains by minimizing the distance between the distribution of latent feature representation. There are various criteria to measure the distribution difference, including Wasserstein distance [69], Kullback-Leibler Divergence [70], etc. Among them, Maximum Mean Discrepancy (MMD) [55] is most frequently adopted in mapping transfer from the surveyed papers. The MMD is calculated as the difference between the mean embeddings of the samples in a reproducing kernel Hilbert space associated with a chosen kernel function. Added to the target loss function, it serves as a powerful tool for comparing the similarity of complex, high-

dimensional datasets using a wide variety of kernel functions.

Previous work has focused on transferred feature extraction/dimensionality reduction using MMD. Wang *et al.* focus more on the subdomain of the same subcategory instead of the alignment of the global distribution between source and target domain [54]. Specifically, they first use the attention mechanism to extract discriminative features that are most related to the fault signal. Then, local MMD is applied to transfer knowledge to adjust the distribution of related subdomains under the same category. Long *et al.* propose their Joint Adaptation Network [56] based on MMD, in which the joint distributions of multiple domain-specific layers across domains are aligned. In addition, an adversarial training version was adopted to make distributions of the source and target domains more distinguishable. Similarly, Long *et al.* adopt multi-layer adaptation and proposed Deep Adaptation Networks (DAN) [57]. The first three convolutional layers are used in DAN models to extract general features. For the last three layers, multi-kernel MMD bridges the cross-domain discrepancy and learns transferable features. Zhang *et al.* propose a Deep Transfer Network in which two types of layers are used to obtain domain invariant features across domains by adding MMD loss. The shared feature extraction layers learn a shared feature subspace between the source and the target samples, and the discrimination layer is then used to match conditional distributions by classifier transduction [58]. Venkateswara *et al.* propose Deep Adaptation Hash network [59], which is fine-tuned from the VGG-F [71] network. Multi-kernel MMD loss is employed to train the Deep Adaptation Hash network to learn feature representations that align the source and target domains.

4) Domain-adversarial transfer

Inspired by Generative Adversarial Networks (GANs) [72], [73], the goal of domain-adversarial transfer is to extract a transferable feature representation that is indiscriminate between source and target domain through adversarial training. Adversarial transfer is primarily concerned with addressing domain adaptation problems.

Soleimani and Nazerfard utilize the GANs framework to perform cross-subject transfer learning [60]. The generator is used to generate samples that are similar to the target

data. Meanwhile, the discriminator distinguishes the fake samples from the target samples. The classifier is trained to discriminate the labeled source data and fake samples to learn generalized features invariant to source and target domains. It is important to note that in real-world applications, training GANs can be unstable due to mode collapse, especially in the case when the source data and target data are unbalanced, and the generator may fail to generate fake samples that can confuse the discriminator. Tzeng *et al.* adopt a domain confusion loss across the source and target domains to learn a domain invariant representation [61]. Ganin *et al.* propose a new domain adaptation architecture by adding a domain classifier after feature extraction layers [63]. A gradient reversal layer is used to ensure the similarity of the feature distributions over source and target domains. Similarly, Ozyurt *et al.* develop a novel framework for unsupervised domain adaptation of time series data by using contrastive learning and domain-adversarial transfer learning [62]. A domain classification loss is applied to extract domain invariant features. The drawback is that the experiments are designed in a way that the source and target data sizes are similar, whereas in practice, the target data is usually much fewer than the source data. Ajakan *et al.* propose a domain adversarial DNN in which a domain regressor is applied to learn a domain invariant feature representation [64]. Tzeng *et al.* use an unsupervised domain adaptation method that combines adversarial learning with discriminative feature learning [65].

D. RELATED LEARNING PARADIGMS

Besides the dedicated transfer learning approaches discussed above, there are methods that represent alternative ways to solve tasks across domains or are complementary to the native transfer learning methods.

- *Multi-task learning* is a machine learning technique where a single model is trained on multiple tasks simultaneously. The idea is to improve the performance of the model by learning a shared representation that captures the features between all tasks. Because the network learns to solve multiple tasks, it may generalize better to new data and tasks.
- *Continuous learning* [74] is a learning process where the model continuously learns new tasks from previous tasks over time without forgetting how to solve previous tasks. To some extent, continuous learning can be seen as a sequential transfer learning process, with the constraint to preserve the performance of the previous tasks, which leads to an accumulation of knowledge over time.
- *Few-shot learning* [75] is a type of machine learning where a model can learn and perform well on a new task with only a limited number of labeled samples. In extreme cases, the model can learn with one label [76] and without any label [77]. Whereas, transfer learning usually involves reusing the model from relevant tasks and continuing training on the target dataset.

- *Domain generalization* [78], [79] focuses on developing a generalized model from one or multiple distinct domains to detect unseen target domain data. The main goal is to overcome the domain shift problem. Domain generalization and transfer learning are both applied to transfer knowledge from source domain to target domain. The major difference between transfer learning and domain generalization lies in the utilization of target domain data. Transfer learning leverages knowledge from source domain and target domain. In contrast, domain generalization solely learns from source domain, without access to the target data.
- *Meta-learning* [80], [81] is known as “learning to learn”. For meta-learning, models are trained on a different set of tasks instead of a set of data in the traditional machine learning setting. In this sense, meta-learning can be seen as a form of transfer learning because it involves transferring knowledge from task to task.
- *Knowledge distillation* [82] effectively learns a small model trained to mimic the behavior of a larger, more complex model. The knowledge learned by the larger model can be transferred to the smaller model, which can then be used for the target task.
- *Self-supervised learning* [83], [84] involves training a model to predict some aspect of the input data without any external supervision. The learned representations can be used for various downstream tasks, including those that involve transferring knowledge from one domain to another.

III. TIME SERIES ANOMALY DETECTION IN INDUSTRY

Time series anomaly detection encompasses statistical techniques to analyze and interpret sequential temporal data. In the context of industrial processes, time series anomaly detection plays a crucial role in automating monitoring, effectively scheduling maintenance, and controlling the efficiency, quality, and performance of these processes. For example, after the detection of an anomaly, another model that captures the relationship between time course and different failure modes or drifts may be exploited for predictive maintenance. For example, in injection molding process monitoring, anomaly detection models are used to analyze recorded sensor data from injection molding machines to detect bad parts and identify the root cause of anomalies [85]. There are two basic ways to detect anomalies: for supervised anomaly detection, labels (normal/abnormal) are needed per time series to build a binary classifier [86]. For unsupervised anomaly detection, an anomaly score or confidence value that is conditioned purely on normal data can be used to differentiate abnormal from normal instances [87] – [88].

A. ANOMALY TYPES

According to the literature [89], an outlier is an observation that deviates significantly from other observations in a way that it is likely that it was generated by a different mechanism. In this survey, we focus on time series data collected from

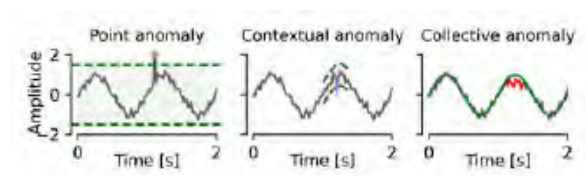


FIGURE 5. Three time series anomaly types. Gray lines represent recorded time series signals, and dashed green lines are *a priori* set thresholds of normal operations. The red dots and the red line indicate anomalies. Point anomalies are single values that fall outside of a pre-set range (left panel). Contextual anomalies are samples that deviate from the current context (middle panel). Collective anomalies are defined as a series of data points that all fall within the range of operation but jointly are not expected (right panel).

machine sensor readings in the context of industrial applications, either univariate (only one variable is recorded over time) or multivariate (several simultaneously recorded measurements). Time series anomalies might occur for various reasons, including internal factors (e.g., temporary sensor error, machinery malfunction) and external factors (e.g. human error, ambient temperature). They can be divided into three categories [32], [33]: point anomalies, contextual anomalies, and collective anomalies. Point anomalies are isolated samples that deviate significantly from the normal behavior of that time series, which can be seen on the left of Fig. 5, e.g., a sudden spike in a pressure reading from a manufacturing machine sensor. These point anomalies can be caused by temporal sensor error, human error, or abnormal machinery operations. Contextual anomalies represent data points that deviate from normal ones only in their current context, and an example can be seen in the middle of Fig. 5. Collective anomalies are a set of data points that in their entirety (but not individually) are abnormal with respect to the entire time series, as shown on the right of Fig. 5.

B. CHALLENGES

Challenges regarding detecting time series anomalies persist due to two specific properties: (1) The *complexity of time series data*. As the automation level of industrial processes and the complexity of industrial systems increases, univariate time series data become insufficient and inefficient in representing any industrial process in its entirety. Hence, more sensors are installed to monitor the whole process, making it necessary to detect anomalies from multivariate time series, which poses particular challenges since it requires consideration of temporal dependencies and relationships between variables and modalities. Many researchers work on discovering generalized patterns from spatial and temporal correlated multivariate time series data. Zhang *et al.* propose a Deep Convolutional Autoencoding Memory network [87], where they build an Autoencoder to capture spatial dependency of multi-variant data using MMD to distinguish noisy, normal and abnormal data. Zhu *et al.* propose an interpretable model agnostic multivariate time-series anomaly detection method

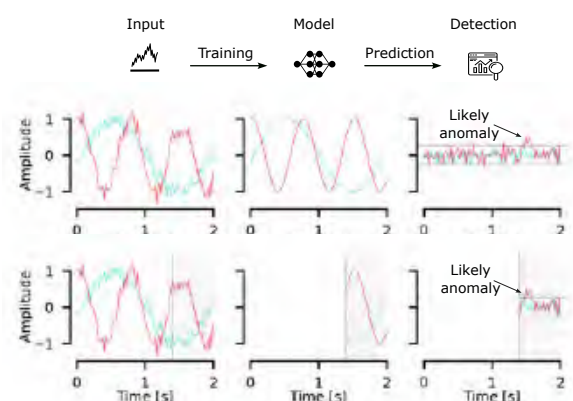


FIGURE 6. Illustration of deep learning-based anomaly detection (top row) with reconstruction-based (center row) and forecasting-based (bottom row) anomaly detection in time series. The first column represents two time series. The second column shows the reconstructed (top) and forecasted (bottom) time series. The third column shows the difference between reconstructed/forecasted time series. Deviations from the reconstructed or forecasted time series are indicative of an anomaly. In deep reconstruction-based anomaly detection, the entire sequence is reconstructed in a decoder-encoder architecture, and the reconstructed sequence is used to compare with the actual sequence. In deep forecasting-based anomaly detection, the end of a sequence is predicted using the start of the sequence and only the forecasted sequence is used to assess the similarity to the ground truth. In this example, the red time series has a likely anomaly at about 1.5 seconds. (Best viewed in color.)

for applications of cyber physical systems [90]. The new method considers both the temporal and feature dimensions through an adaptive mask based series saliency module to produce accurate anomaly detection results and reasonable interpretations in the form of a mask matrix. (2) The *dynamic variability in industrial processes*. Industrial processes often have high dynamic variability and can be affected by a wide range of conditions, such as changes in temperature, pressure, and humidity. These conditions can cause fluctuations in the process outputs, which leads to data shift and domain shift. This can make it challenging to detect anomalies and maintain control over the industrial process.

C. ANOMALY DETECTION METHODS

Time series anomaly detection has been investigated for decades, and various types of methods have been proposed [91]. This paper exclusively discusses the time series anomaly detection techniques using deep learning, leveraging its robust representation learning capabilities. Current deep learning methods can be mainly divided into reconstruction-based, forecasting-based, and other methods. Fig. 6 illustrates the two main methods. In deep reconstruction-based anomaly detection, the reconstructed sequence is used to compare with the actual sequence. Differently, in deep forecasting-based anomaly detection, only the forecast sequence is used to assess the similarity to the ground truth.

a: Reconstruction-based methods

Reconstruction-based methods aim to learn the data distribution of the normal time series and differentiate the abnormalities from the normal ones by computing the reconstruction errors. Audibert *et al.* propose a fast and stable method – Unsupervised anomaly detection for multivariate time series [92], based on adversely trained autoencoders. The encoder-decoder architecture within an adversarial training framework combines the advantages of autoencoders and adversarial training while compensating for the limitations of each technique. After training two autoencoders, the anomaly score is defined by balancing the reconstructed errors from the two autoencoders with two hyperparameters. However, the challenge arises in selecting these two hyperparameters of the anomaly score when the testing dataset is unavailable.

Malhotra *et al.* also formulate an anomaly score based on reconstruction error [93]. They first train the LSTM encoder-decoder model to reconstruct the normal time series. Subsequently, they leverage the reconstruction errors to calculate the probability by using Maximum Likelihood Estimation to detect a specific point within a time series as an anomaly. They set a window to detect anomalies. The window will be labeled as anomalous if the probability exceeds a threshold. Similarly, Wei *et al.* also propose an LSTM-based encoder-decoder model to detect multivariate time series sequences based on the reconstruction error [94]. The major difference is the anomaly detection criteria. They assume the reconstruction error of train/test data follows the normal distribution and detect anomalies by using the 2-sigma rule of the normal distribution as a threshold. Zeng *et al.* propose an adversarial transformer structure to detect multivariate time series anomalies effectively [95]. Here, two-stage adversarial training is applied for the transformer. In the first stage, two transformers are trained by minimizing the reconstruction error to capture the temporal trends in the time series. In the second stage, the reconstruction error serves as prior knowledge in the adversarial training process, enabling the model to distinguish anomalies from normal time series. Then, an anomaly score is defined by combining the anomaly probability and reconstruction error. Again, a threshold has been chosen to differentiate anomalies from normal ones.

GANs, as effective unsupervised learning methods, have been used in time series anomaly detection. Anomaly detection methods based on GANs focus on extracting features by adversarial training on normal samples. Consequently, features from the abnormal samples diverge from those of the normal ones, reflecting in reconstruction error and discrimination value. Li *et al.* use LSTM-RNN as a base model for building generator and discriminator in GAN [96]. The proposed framework considers multiple variables to capture the temporal correlation of multi-time series distributions. Additionally, they proposed a novel anomaly score, which can detect anomalies through discrimination and reconstruction. More specifically, the score is a combination of the reconstruction difference between generated data and original data and the discrimination results from the discriminator. Simi-

larly, Niu *et al.* and Bashar *et al.* both propose an LSTM-based VAE-GAN for time series anomaly detection, where LSTM networks are used as the generator, and discriminator [97], [98]. When it comes to anomaly scores, setting an optimal threshold is usually a critical step. However, using a small portion of the test set to decide the optimal threshold may not be practical in real-world scenarios [97]. Additionally, it is important to note that the method has been only tested for point anomaly detection, further investigation is required when they are applied to detect other anomaly types.

b: Forecasting-based methods

Forecasting-based methods predict the value of the following timestamps and predict temporal anomalies according to the prediction error. Kim *et al.* propose a forecasting-based unsupervised time-series anomaly detection method using transformer architecture [99]. The idea is to train a transformer-like model by forecasting a fixed-length time series based on the previous timestamps. The trained model is used to predict time series with an anomaly score such that an instance where the anomaly score is larger than a static threshold is defined as an anomaly. A dynamic thresholding technique is also mentioned but not explicitly discussed in the paper.

Deng and Hooi propose a novel attention-based graph neural network approach [100] that learns a graph of dependence relationships between multi-variant time series signals by forecasting the behavior based on past time series. Then, a graph deviation scoring is defined for each sensor to detect and explain anomalies. Tang *et al.* propose an interpretable multivariate time series anomaly detection method based on graph neural networks and gated recurrent units [101]. The feature representation is learned through forecasting the future time series segment. An abnormal score is set for each time series to detect anomalies. The feature embedding is then used for 2D visualization through t-SNE plots to interpret the clusters within time series from different sensors.

c: Other methods

Ding *et al.* propose a joint network to integrate the advantages of reconstruction and forecasting/prediction [102]. First, they propose a multimodal graph attention network to tackle the spatial-temporal dependencies for multimodal time series. Further, they optimize the reconstruction and prediction modules simultaneously to predict anomalies. Himeur *et al.* take advantage of annotated data and directly use a DNN as a classifier to classify normal and abnormal energy consumption types [103]. The enormous imbalance of real anomaly patterns is one concern in the approach. Thus, a normalized technique of power consumption data is applied to deal with this problem. The normalized data represent the difference in power consumption rates of each current time sample and the previous one. It can provide information on how fast the consumption reacts to the time evolution. However, any further evaluation of this technique is not discussed, and it is still an open problem regarding anomaly detection for other datasets. Yang *et al.* propose a contrastive learning structure with dual

attention to learn a permutation invariant representation of the data with superior discrimination characteristics between normal points and anomalies [104]. Unlike most reconstruction-based models, their model is a self-supervised framework based on representation learning. The new method achieves state-of-the-art comparable performance on six multivariate and one univariate time series anomaly detection benchmark datasets. However, the extensive framework with two multi-head-attention blocks may be prone to overfitting, especially since no training details, only evaluation details are disclosed.

In principle, these anomaly detection approaches are applicable to all types of anomalies. Reconstruction-based methods are typically applied to the entire or a portion of the time series. Long-time series are commonly segmented into subsequences using a predefined sliding window. In the case of detecting context/collective anomalies, the reconstructed loss of the time series sequence is evaluated within the predefined sliding window, if the reconstruction loss is larger than an acceptable threshold, then that time series sequence is classified as an anomaly. In the case of point anomalies, reconstruction is performed at each single time stamp, akin to a regression problem, and then the reconstruction loss of each single timestamp is evaluated to determine whether the single timestamp is anomalous or not. It is also applied to forecasting-based anomaly detection methods, instead of computing reconstruction error, forecasting-based methods predict the value in the next time stamp for point anomalies or the next time series sequence for context/collective anomalies. The anomalies will be detected based on the deviation between the predicted value and the normal value. Other anomaly detection approaches usually combine the reconstruction-based and forecasting-based methods.

To sum up, these methods are applicable to each type of anomaly. However, the effectiveness of these anomaly detection approaches may vary depending on the anomaly detection tasks at hand, which are characterized by the granularity at which the time series data is observed and analyzed.

IV. INDUSTRIAL APPLICATIONS

A. OVERVIEW

Deep transfer learning techniques have gained prominence in computer vision and natural language processing, primarily due to the abundance of available datasets. However, their adoption in the context of industrial time series data has been comparatively limited. This hesitancy can be attributed to the limited public availability of such datasets and the unique domain-specific characteristics they possess, which complicate generalized advancements. Encouragingly, there has been a recent uptick in the application of deep transfer learning for anomaly detection within the industry such as fault diagnosis [105], quality management [106], manufacturing process monitoring [85], network/software security [107], and infrastructure monitoring [108]. These can be mapped onto the core industrial domains of manufacturing process and infrastructure monitoring, predictive maintenance, and



FIGURE 7. Overview of Sankey diagram of transfer learning problem setting, deep transfer learning approach categories, and label availability in the surveyed industrial domains.

energy management. Table 2 presents a compact comparison of the related works using deep transfer learning approaches to solve these tasks.

Fig. 7 illustrates the Sankey diagram of the connections between industrial applications and the deep transfer learning approaches based on our literature survey. The diagram shows every path that connects the four dimensions of the methodology-problem-landscape within the surveyed literature. The broader the path is, the more papers are related to that element. The goal is to give an overview of how deep transfer learning is applied to industrial problems in the recent literature and specifically show with these four dimensions: (1) which deep transfer learning approaches are actually used in practice; (2) what the main industrial domains for time series anomaly detection are; (3) what deep transfer learning category these domains belong to; (4) what labels are available in source and target domain.

Key observations from Fig. 7 are: (1) parameter transfer is much more frequently used than any other deep transfer learning approach across all surveyed industrial applications since fine-tuning a pre-trained model on target data is more straightforward to implement by taking advantage of the pre-trained model on the source dataset and usually without fundamental modification on the model architecture. It is noteworthy that instance transfer and adversarial transfer do not appear in the diagram. Apparently, these two deep transfer learning approaches are not considered effective in time series anomaly detection tasks in industry. The difficulty lies in implementing and training these scarcely researched approaches in the industrial field, as indicated by the findings. (2) Hybrid approaches of parameter and mapping transfer can be seen in predictive maintenance. (3) Most industrial applications use inductive transfer learning, indicating they focus on leveraging labeled source and target data to solve the target task, i.e., use supervised learning.

B. MANUFACTURING PROCESS MONITORING

Manufacturing process monitoring is crucial to ensure high-quality products and low rejection rates. For example, in

TABLE 2. A compact overview of industrial applications that used deep transfer learning for time series anomaly detection.

Reference	Industrial task	Industrial domain	Deep transfer learning approach	Transfer problem setting	Learning	Deep learning framework	Source type	Source label	Target label
[11]	Industrial metal forming anomaly detection	Predictive maintenance	Parameter transfer	Inductive	Inductive	CNN	Multiple	✓	✓
[12]	Manufacturing systems anomaly detection	Predictive maintenance	Parameter transfer	Inductive	Inductive	U-Net	Multiple	✓	✓
[13]	Casting process fault diagnosis	Predictive maintenance	Parameter transfer	Inductive	Inductive	SAE	Multiple	✓	✓
[14]	Roller bearings fault detection	Predictive maintenance	Mapping transfer	Transductive	Transductive	Auto-encoder	Multiple	✓	✓
[15]	Industrial control systems anomaly detection	Predictive maintenance	Parameter transfer	Inductive	Inductive	ResNet8	Single	✓	✓
[16]	Service elevator fault detection	Predictive maintenance	Parameter transfer	Inductive	Inductive	CNN, RNN	Multiple	✓	✓
[19]	Nuclear power plants fault detection	Predictive maintenance	Parameter transfer	Inductive	Inductive	CNN	Multiple	✓	✓
[109]	Building energy systems fault diagnosis	Predictive maintenance	Parameter transfer	Inductive	Inductive	CNN	Multiple	✓	✓
[110]	Press machine production prediction	Predictive maintenance	Parameter transfer	Inductive	Inductive	CNN	Single	✓	✓
[111]	Wind turbine fault detection	Predictive maintenance	Parameter transfer	Inductive	Inductive	CNN	Single	✓	✓
[112]	Industrial machine operating fault detection	Predictive maintenance	Parameter transfer	Inductive	Inductive	CNN, LSTM	Single	✓	✓
[9]	Machine turning operations classification	Manufacturing process monitoring	Parameter transfer	Inductive	Inductive	VGG, ResNet	Single	✓	✓
[10]	Injection molding process quality control	Manufacturing process monitoring	Parameter transfer	Inductive	Inductive	FCN	Multiple	✓	✓
[85]	Injection molding process quality control	Manufacturing process monitoring	Parameter transfer	Inductive	Inductive	FCN	Single	✓	✓
[113]	Injection molding process anomaly detection	Manufacturing process monitoring	Parameter transfer	Inductive	Inductive	FCN	Multiple	✓	✓
[114]	Injection molding process anomaly detection	Manufacturing process monitoring	Parameter transfer	Inductive	Inductive	FCN	Multiple	✓	✓
[115]	Aluminum gravity die casting quality prediction	Manufacturing process monitoring	Parameter transfer	Inductive	Inductive	FCN	Single	✓	✓
[20]	Agriculture/manufacturing systems anomaly detection	Manufacturing process monitoring	Parameter transfer	Inductive	Inductive	LSTM	Single	✓	✓
[116]	Manufacturing testbeds anomaly detection	Manufacturing process monitoring	Parameter transfer	Inductive	Inductive	LSTM, RNN	Single	✓	✓
[117]	Industrial metal (pump) forming anomaly detection	Manufacturing process monitoring	Parameter transfer	Inductive	Inductive	LSTM	Single	✓	✓
[118]	Industrial control systems anomaly detection	Manufacturing process monitoring	Parameter transfer	Inductive	Inductive	Auto-encoder	Single	✓	✓
[21]	Petrochemical production process anomaly detection	Energy saving	Parameter transfer	Inductive	Inductive	LSTM, CNN	Single	✓	✓
[119]	Electricity consumption anomaly detection	Energy saving	Parameter transfer	Inductive	Inductive	FCN	Single	✓	✓
[120]	Building's energy consumption anomaly detection	Energy saving	Parameter transfer	Inductive	Inductive	AlexNet-40	Single	✓	✓
[121]	Power consumption anomaly detection	Energy saving	Mapping transfer	Transductive	Transductive	DNN	Single	✓	✓
[122]	Building's power consumption anomaly detection	Energy saving	Parameter transfer	Inductive	Inductive	LSTM	Single	✓	✓
[23]	Aircraft flight anomaly detection	Infrastructure facilities monitoring	Parameter transfer	Inductive	Inductive	LSTM	Single	✓	✓
[108]	Anomaly identification for bridge groups	Infrastructure facilities monitoring	Parameter transfer	Inductive	Inductive	CNN	Single	✓	✓
[123]	Network intrusion detection	Infrastructure facilities monitoring	Parameter transfer	Inductive	Inductive	CNN, LSTM	Single	✓	✓
[124]	Building occupation detection	Infrastructure facilities monitoring	Parameter transfer	Inductive	Inductive	CNN	Single	✓	✓
[18]	Building occupation detection	Infrastructure facilities monitoring	Parameter transfer	Inductive	Inductive	CNN	Single	✓	✓

injection molding machines, sensors are installed to detect molding conditions in the cavities, such as cavity pressure and temperature. These signals are used to analyze particularly the mold filling and solidification process for each produced part. Such cyclic processing data can also be seen in metal machining (cutting force signal) or joining of parts (joining force signal). Currently, parameter transfer is predominantly used for manufacturing processes [10], [11], [20], [85], [113], [114], [116], [117], [123].

Park *et al.* propose a transfer learning technique to detect time series anomalies for different industrial control systems [118]. First, they apply principal components analysis to reduce the dimension of source and target data. A DNN model is then trained on the compressed source data, and after a reasonable mapping algorithm is adopted to map the features of source to target domain, the pre-trained model is further trained on the target data. The model achieves good performance even when a model is retrained with only a proportion of target data. For the experiments, they only test on two comparatively larger datasets and fail to show that the transferred model performs better than the one without retraining for one dataset. Further investigation needs to explain the negative transfer. Additionally, even when they only take a small proportion of target data for transfer learning purposes, the sample size still exceeds 5000, which exceeds most industrial applications. Abdallah *et al.* apply parameter transfer to monitor the operation status of manufacturing testbeds with vibration sensor data [20], [116]. Hsieh *et al.* transfer knowledge across three chambers in a production line to detect anomalous time series data [123]. Results show reduced training time and improved detection accuracy through transfer learning. In injection molding, parameter transfer is applied to transfer the knowledge from one or more source domains to solve tasks in a target domain [10], [113], [114]. Specifically, they employ simple fully connected neural networks and transfer knowledge from one product to another by freezing the first few layers and fine-tuning only the last few layers. Instead of evaluating the time series data directly from sensors, they represent the industrial process by the parameters of the machine settings. However, they can still provide useful insight for the case of the time-series data. Tercan *et al.* build a bridge between simulated data and real data using parameter transfer in injection molding [85]. Here, a fully connected neural network is trained on simulated data and then partially or fully reused to further train on real data. Results show that the transferred model performs better than a network trained from scratch on real experimental data. In manufacturing processes, a simulation model/process hence can play a critical role, but deeper analysis is needed to further understand and reduce the gap between simulation and real data. Additionally, Lockner *et al.* explore the impact of transfer learning with varying amounts of source data and assess how performance is influenced by different configurations of frozen layers [114]. Maschler *et al.* compare different DNNs for anomaly detection tasks on metal forming datasets [11]. Further, they propose a deep transfer learning frame-

work aiming to transfer knowledge between tasks. However, the proposed architecture is not validated by experiments. Later, Maschler *et al.* apply continuous learning on the same dataset by transferring knowledge from several source tasks to a target task to train a deep learning algorithm capable of solving both source and target tasks [117]. Specifically, they use regularization approaches using altered loss functions to solve related tasks that appear best suited.

C. PREDICTIVE MAINTENANCE

Predictive maintenance aims to predict the necessity of maintenance before production is negatively impacted by a failure. Tasks involve monitoring equipment to anticipate maintenance requirements (i.e., predict probable future failure) to optimize maintenance schedules [124]. Time series anomaly detection is often used in respective systems to identify abnormal behaviors in operation that may indicate the need for maintenance, such as increasing noise, vibrations, etc.

Mao *et al.* use mapping transfer with a Sparse Auto-Encoder (SAE) for motor vibration anomaly detection [14]. A transformation from the source and target data to a common latent feature space is learned with MMD loss to make the feature distribution of two domains as identical as possible. Similarly, Wen *et al.* also use mapping transfer with an SAE architecture for fault detection of rotation bearings, using an MMD regularizer to extract a common feature representation [125]. Subsequently, they propose a new MU-Net architecture to detect multivariate time series anomalies [12]. First, they pre-train a U-Net [126] on a large time series dataset for an anomaly detection task. Then, they propose a new model MU-Net, which is built upon U-Net. In MU-Net, each channel can leverage a pre-trained U-Net through fine-tuning to transfer knowledge for multivariate time series anomaly detection.

In a different application, parameter transfer is used to predict the remaining useful life for tools in manufacturing [127]. An SAE network is first trained to predict the remaining useful life of a cutting tool on retrospectively acquired data in an offline process. The trained network is then transferred to production with a new tool for online remaining useful life prediction. The result shows that transfer-learning based hybrid deep learning significantly reduces the training time and is highly suitable for real-time industrial fault diagnosis/prediction in various environments. Similarly, parameter transfer is implemented to reduce the gap between different industrial environments [13], [112]. Xu *et al.* use a stacked SAE to extract general features from source data and a digital-twin-assisted fault diagnosis approach is presented to transfer knowledge from virtual space to physical space for real-time use [13]. Here, a DNN model is first fully trained in virtual space and then migrated to the physical space using parameter transfer for real-time use.

The above-mentioned literature proves that deep transfer learning is a research field that could simplify the life cycle of predictive maintenance systems and facilitate DNN model reusability by reducing the required data and training time, helping adapt them to solve similar tasks.

D. ENERGY MANAGEMENT

Energy management deals with systems that detect abnormal excessive consumption caused by end-users' unusual behavior or malfunction of faulty devices or systems [120]. The goal is to develop automatic, quick-responding, accurate, and reliable fault detection to save energy and build environmentally friendly systems. Energy anomaly detection systems monitor data during energy generation, transmission, and utilization, to ensure normal energy consumption.

Xu *et al.* design a cluster-based deep adaptation layer to improve a deep adaptation network, effectively reducing the mismatch in transfer learning of spinning power consumption anomaly detection [121]. The basic architecture consists of five convolutional layers and three fully connected layers. The weight parameters of the convolutional layers are shared between source and target domains. The cluster-based deep adaptation layer is designed across the feature layers of two networks to cluster feature representations of the source and target domains respectively. The proposed method shows superiority over fine-tuning and DAN because the adaptation layer can minimize the distance between the nearest neighbor clusters across the source and target domains to match the most similar distribution of feature representations. It is important to note that the anomalies are defined and tagged by human experts as different types, thus the problem becomes a classification task. However, in real-world industrial applications, it's almost impossible to enumerate unknown anomaly types because of the highly dynamic environment. Liang *et al.* successfully build an electricity consumption time series anomaly detection method in aluminum extrusion [119]. Parameter transfer is applied to transfer domain knowledge from another data-sufficient domain. First, they train on sufficient extruding machine data in an unsupervised way and then use only a few data samples from different extruding machines to adapt the model by transfer learning. It is important to note that when the target data is already sufficient, transferring knowledge can be detrimental as it can decrease prediction accuracy on the final task. Copiaco *et al.* aims to detect anomalies for building energy consumption via transfer learning from pre-trained CNN models [128]. First, they convert 1D time series signals to 2D image representations. These serve as inputs for pre-trained vision models to capture inherent spatially invariant features. In the end, a SVM is applied to classify anomaly types. The SVM classifier obtained optimal results when operating upon a pre-trained ALEXNet model with normalized grayscale graphical representations. However, a deeper discussion regarding the effect of the different pre-trained models is not presented. Additionally, converting 1D time series to 2D images by creating a matrix representation of the sensor readings may lead to information loss during the transformation process, which should be further investigated.

E. INFRASTRUCTURE FACILITIES MONITORING

Infrastructure facilities monitoring refers to monitoring and maintaining the conditions of infrastructure facilities, such

as bridges, buildings [129], and networks. This can include detecting potential problems or failures. The goal is to minimize the impact of failures on the public or the environment. This application commonly uses parameter transfer to transfer knowledge from facility to facility to take advantage of similar data and tasks.

Dhillon *et al.* present a parameter transfer approach towards building a network intrusion detection system based on CNN and LSTM [22]. Specifically, They extract and learn patterns by mapping the input data into a lower dimensional representation by convolutional layers. Then, they employ the LSTM layer to enhance learning and recognizing patterns across time. In the end, a fully connected layer is used as a classifier to predict normal and malicious data. To do the parameter transfer, they reuse the model architecture and freeze most weight parameters for the target domain so that they do not need a large training dataset to retrain the model. However, they do not mention implementation details, like which layers are frozen in the transfer learning stage. Observing how transfer learning performs with different frozen layers would be interesting. Pan *et al.* apply parameter transfer to fully use the similarity of the anomalous patterns across different bridges [108]. They train a CNN model on one bridge data, then transfer the knowledge obtained by the CNN model to a small part of the target data. They update the last three fully connected layers while keeping the convolutional layers intact. The experimental results show transfer learning achieves higher accuracy anomaly detection across bridges. Weber *et al.* takes advantage of simulation data by training on synthetic environmental data, then fine-tunes the pre-trained model and transfers the knowledge from simulation data for real-time online building occupancy detection [17]. Although the results show the effectiveness of transfer learning, the availability, and reliability of the simulation data for other industrial applications is still an open issue. Sayed *et al.* adapt parameter transfer using pre-trained CNN models, such as AlexNet and GoogLeNet, pre-trained on ImageNet [18]. The pre-trained model is then further used for downstream tasks. The results show the pre-trained models outperform their customized CNN model, which is not pre-trained. However, it is important to note that the transfer effect may not be entirely convincing due to the fact that the customer CNN model is not pre-trained on the same dataset.

F. APPLICATION-INDEPENDENT CONSIDERATIONS

Data scarcity, as well as domain shift, stand out as the two main common problems independent of the industrial field of application. The same problems have originally prompted the use of transfer learning in general, and respectively, general techniques are applied widely across domains. Regarding data scarcity, this un-surprisingly involves leveraging pre-trained models as a starting point for further training. Regarding domain shift, mapping transfer and parameter transfer are the most often-used approaches. Unlike parameter transfer, mapping transfer incorporates the source and target data in the training process. Instance transfer and domain-adversarial

transfer learning were not employed in the surveyed literature – researchers seem to not see huge value in these methods for the surveyed fields.

Another common aspect across time series anomaly detection applications is the choices of model architecture to facilitate the training process by capturing temporal dependencies and recognizing patterns over varying time scales: Favourite architectures include CNNs, LSTMs, and auto-encoders that have sets of assumptions (inductive biases) about the data they analyze that make them excel in understanding the sequential nature of time series data. CNNs assume local (in time) connectivity, stationary statistics, and hierarchical structure, and induce certain translation-invariance. LSTMs are still given preference in many applications over the more modern deep learning architecture of choice for sequence learning, the transformer. The reason is their stronger inductive bias, leading to less data (and compute) hunger. Both CNN and LSTM networks can be built as e.g. classifiers, but also auto-encoders. These latter architectures have the advantage of learning low-dimensional representations of the high-dimensional time series with minimal loss of signal in an unsupervised way. The analysis of the data in the low-dimensional latent space facilitates anomaly detection. The concrete choice of architecture does not depend on the field of application but on the data and task, and thus the most suitable inductive bias.

As an interim conclusion, the most striking application-independent finding is that across the surveyed literature, predominantly simple, tried-and-tested design patterns for transfer learning are used in industry. The field of deep transfer learning would offer a much wider variety of approaches.

V. DISCUSSION

A. POTENTIAL

The automation of industrial process monitoring stands as a transformative step toward increasing efficiency and optimizing quality. While standard deep learning training is sufficient in discerning intricate patterns from vast datasets, its application in the dynamic industrial landscape is not without challenges. Chief among them is the impracticality of continuously obtaining large-scale labeled data to train models afresh for every nuanced variation in processes. Deep transfer learning has shown promise with its adaptive capabilities. By mitigating the need for extensive labeled data and eliminating the necessity to train models from scratch for every distinct setup. However, adopting deep transfer learning beyond simple parameter transfer is still a challenge.

B. CHALLENGES

a: Domain shift

Different from the i.i.d assumption in most machine learning problem settings, many industrial processes suffer from substantial domain shifts due to dynamic changes in industrial settings, e.g., change of products or measuring sensors. Domain shift lies at the heart of the deep transfer learning problem. Particularly, the dynamic changes in many industrial

processes, up to an apparent dissimilarity of source and target data, make the transfer learning task particularly challenging. Here we list the key challenges associated with domain shift:

- *Covariate shift* occurs when the marginal distribution of the features changes from source domain to target domain. The distribution mismatch poses challenges in transferring the knowledge from source to target domain.
- *Concept shift* refers to the changes in the relationships between features and labels. The relationship can change from source domain to target domain, leading to bias and error in the model.
- *Label shift* refers to the label distribution in the target domain that can be different from the source domain, whether the marginal distribution changes or not.

b: Label availability and reliability

Deep transfer learning is built upon deep learning, which usually requires a large amount of labeled data, the more data a model has available for learning, the better it can generalize to new examples. In real-world industrial time series anomaly detection tasks, collecting data is probably easy, but collecting labels is much more expensive and time-consuming, sometimes prohibitively so, leading to the unavailability of sufficient labeled data. Self-supervised learning can be used to re-label a large amount of unlabeled data and thus anomaly detection models usually need to learn in an unsupervised or semi-supervised mode [130]. In industrial cases, another significant concern is to ensure the data quality. Due to the high cost associated with obtaining reliable and precise labels, usually self-supervised learning is applied to create pseudo labels or relabel the unlabeled data, thus facilitating the transfer learning process. Additionally, a data-centric process with humans in the loop can be involved in improving label reliability. However, unreliable labels can still affect the transfer learning training process.

c: Missing relevant data information

Missing relevant data poses significant challenges for transfer learning since it can affect the model's ability to generalize and transfer knowledge from source to target domain.

- *Imbalanced data*: Even if the labels can be collected, anomalies can be extremely rare by design, which poses the risk of training with extremely imbalanced data. A practical problem for anomaly detection in industry is the extremely imbalanced data distribution, in which normal samples dominate in data and abnormal samples only share a small percentage in the whole dataset. Prior research has proven that the effect of class imbalance on classification performance by using deep learning is detrimental [131]. However, most research studies still ignore such problems, which can result in poor performance regarding the minority class, i.e., abnormal data are misclassified as normal.

- *Information loss*: missing data can lead to lost important features. For example, some information that has a significant effect on the process from case to case is not even recorded or is too complex to record (i.e. part geometry, machine geometry, or environmental conditions in injection molding processes).

Various approaches have been developed to address these challenges to reduce the domain gap between the source and target domains, aiming at mitigating domain shift. These techniques involve domain generalization, contrastive learning, and adversarial examples. The domain shift problem is far from being solved. To tackle this problem, transfer learning requires a deep understanding of the target data's characteristics and appropriate transferable strategies to effectively bridge the gap between the source and target domains.

d: Effectiveness of deep transfer learning

The general effectiveness of deep transfer learning is limited by the difficulty of determining which knowledge or to what extent the knowledge should be transferred from source to target task. Unlike natural language processing, pre-training a language model on a large corpus of text data can help the model learn the statistical patterns and semantic and syntactic representations of words and sentences, which can be used for new natural language processing tasks with a few or even without data. Due to data privacy, large available public datasets usually do not exist for industrial time series, or they can not be used because of a large domain gap between different datasets and tasks. In this case, transferring all of the knowledge may not be beneficial, as it may be irrelevant. In the worst case, this can lead to negative transfer [28], [132], in which the extracted knowledge harms the new task-learning. This requires assessing how source and target tasks are related, carefully selecting the knowledge to be transferred, and selecting the proper means to implement this transfer. Glorot *et al.* attempt to analyze and quantify the gained knowledge from source to target domain [133]. For example, they define transfer error, transfer loss, transfer ratio, and in-domain ratio, which provide metrics to interpret the transferring performance.

C. DIRECTIONS FOR ANOMALY DETECTION SOLUTION DESIGN

a: Data preprocessing

How data preprocessing should be conducted is an open question. For industrial applications, some researchers contend that using raw time series data directly as input for training may not be the most efficient. Hence, they propose deriving or selecting features from time series data by statistical methods or human experience. This can significantly decrease the complexity of the dataset. On the other hand, this crops a lot of potentially useful information, e.g., the time series trend. To reduce the dimensionality, some researchers use machine parameters as features in the manufacturing process instead of the processing data collected by sensors [10], [85],

[113], [114]. Others try different transformations of raw time series data, a common way being to transform 1D time series data to 2D image data [9], [15], [112] or transforming time domain signals otherwise into the frequency domain [39]. However, as large-scale computation power and storage become cheaper and more accessible, it is becoming increasingly common to use deep learning techniques to process time series data directly [11], [13].

b: Data augmentation by generative AI

Data augmentation is useful for deep learning models because it can help to prevent overfitting. For deep transfer learning, when a model becomes too closely adapted to the specifics of the source domain, it may not be able to generalize well to some examples in the task domain. One important technique is to acquire effective synthetic data, e.g., using a simulation process or model to explore potential anomalous conditions by simulating industrial processes under parameters that cannot yet be experienced in the real world. High fidelity and reliable simulation data can provide training data at low cost and mitigate the problem of insufficient samples for deep transfer learning [13]. Another way to generate effective synthetic data is to use generative models, such as GANs. GANs are only trained on normal data to generate indistinguishable normal samples so that abnormal samples can be distinguished during the testing stage of the overarching anomaly detection system, as they deviate from the normal data distribution [134]. To increase the number of anomalous samples and thus the robustness of the anomaly detection model, the technique of adversarial perturbation known from computer vision [135] can be used.

c: Dealing with data imbalance

DNNs perform well when they are trained on balanced datasets. However, in practice, it is difficult to get sufficient anomalous data for anomaly detection tasks. For example, the manufacturing process is usually in a healthy state due to the pre-designed and optimized operation. Several ways exist to address the imbalanced dataset for time series anomaly detection. One way is to oversample the minority class, e.g., by randomly replicating samples from the minority class to equalize the number of samples from each class in each batch. The Synthetic Minority Over-sampling Technique is an advanced method that creates synthetic samples to force the decision region of the minority class to become more general [136]. This technique is widely used in anomaly detection tasks in industry [137], [138]. Apart from oversampling, resampling strategies are frequently used to assign a higher probability to abnormal samples and evenly select the same amount of samples from both classes in each batch. Moreover, a weighted loss can be implemented to balance the loss between the abnormal and normal class in supervised anomaly detection [131].

D. DIRECTIONS FOR DEEP TRANSFER LEARNING IMPLEMENTATION

a: When shall deep transfer learning be used?

(1) Limited data availability: It poses a significant challenge in machine learning, particularly when aiming to train models for specific tasks. Pre-training a model on a larger or more diverse dataset, even if unrelated to the specific task at hand, enables the acquisition of generalizable features and representations. These generalized features, learned from a broader context, can then be effectively transferred to analyze the target domain with limited data. This can effectively provide a practical solution to the challenges posed by data scarcity. (2) Similar domains: Deep transfer learning is well suited when tackling source and target domains with a high degree of similarity. In such instances, knowledge can be derived either from a model pre-trained on a similar dataset or one trained on both source and target data. In both cases, the model can efficiently transfer relevant features and representations within the domain, facilitating a more robust adaptation to the target dataset, and ultimately optimizing the model's ability to discern and detect patterns within the target domain. (3) Limited resources (encompassing both time and computational power): When faced with resource constraints, it is recommended to employ parameter transfer, especially if a pre-trained model is readily available. As described in [34], the transfer might improve learning in three distinct ways: (a) a higher performance at the very beginning of learning, (b) a steeper slope in the learning curve, or (c) a higher asymptotic performance. Parameter transfer leverages the learned parameters and weights of a pre-trained model, often trained on a larger dataset. By doing so, the resource-intensive process of training a model from the ground up is circumvented, and the computational burden is significantly alleviated.

b: When not to use deep transfer learning?

(1) Irrelevant data: If the target data is vastly different from the source data, deep transfer learning may not be appropriate, sometimes even leading to negative transfer. For example, if one wants to train a model for natural language processing on a new dataset, using a pre-trained model that has been trained on image data may not yield meaningful results. This is due to the vast dissimilarity in data modalities and features between images and text. (2) Task-specific models: In scenarios where the target task is well-defined and specific, and pre-trained models do not align closely with the task requirements, it is usually more effective to build a task-specific model from scratch. (3) High domain shift: If there is a large difference between the source and the target domain, deep transfer learning may not be effective. This can happen when the data distributions, features, or labels are vastly different. (4) Abundance of labeled data available for target task: If there are enough data for the new task, it may be more effective to train a model from scratch [119].

c: What model architecture to choose?

We recommend choosing the model architecture mainly based on data size and label availability, starting from a relatively small network and moving gradually to more complex

DNNs. CNNs also effectively extract time series features [19], [111]. For semi-supervised settings, CNN-based auto-encoders are trained to reconstruct the original data [110]. It is important to effectively capture the temporal dependencies and extract features of time series data. LSTMs are extensively employed for this purpose, as they excel in detecting temporal dependencies in time series data [21], [23], [112].

Exploring hybrid architectures that combine the advantages of CNNs, RNNs, and LSTMs for tasks involving both spatial and temporal dependencies can be beneficial. Cao *et al.* propose a multi-head CNN-RNN architecture for multi-time series anomaly detection [16]. A CNN is used to extract meaningful features from raw data and then an RNN is applied to learn temporal patterns simultaneously. Similarly, Dhillon *et al.* utilize LSTM layers to model the time series signals after obtaining the features from a CNN. An alternative way to benefit from different models is the use of ensemble approaches, combining the strengths of different model architectures to enhance performance, especially in situations where the target task requires capturing diverse features. In the future, we expect more applications to use transformer-based approaches as pre-trained models become available and public datasets get open-sourced.

d: Beyond transfer learning

Foundation models like SAM [139] or others, using for example transformer architectures [48] or diffusion models [140], demonstrate emerging properties such as in-context learning [53] and complex cross-modality conditioning. This is achieved by training complex and often auto-regressive models with massive amounts of data, although the precise mechanisms that lead to this are not well understood. Some of those models generalize to new settings and tasks, without an explicit element of transfer learning. Thus, the application of foundation models in industrial time series analysis has the potential to reduce and eventually eliminate the need to explicitly account for changes in the domain within the modeling, by instead having the foundation model provide the transfer capability (see examples in Sec. VI-b). To not only detect anomalies but also identify failure modes, analyze root causes, and elicit an appropriate intervention, AI systems must implicitly or explicitly model causal relations. Counterfactual inference incorporates causal relations between observations and interventions, which allows predictions of outcomes never seen during training [141].

Another aspect of deep learning implementations is the limited computing power of hardware platforms, such as embedded systems in industry. Sensor data are typically acquired using resource-constrained edge processing devices that struggle with computationally intensive tasks, especially when training a DNN model. Federated learning stands out as a leading solution, with its ability to utilize data while preserving privacy [142], [143]. The technology enables a more collaborative approach to ML while preserving user privacy by storing data decentralized on distributed devices rather than on a central server. Combining deep transfer

learning with federated learning is a promising and powerful combination in the abovementioned industrial applications.

VI. CONCLUSIONS

In this survey, we presented a comprehensive overview of deep transfer learning by defining transfer learning problem settings and categorizing the state-of-the-art deep transfer learning approaches based on the surveyed papers. Then, we review and emphasize on investigating deep transfer learning approaches for time series anomaly detection in different industrial settings. Equipped with this foundation, we selected representative examples of the landscape of fielded applications to provide practitioners with a guide to the field and possibilities of industrial time series anomaly detection.

The main finding of this survey is that only a limited variety of deep transfer learning methods are employed in anomaly detection in industrial time series analysis – mainly simple ones. Almost all applications employ parameter transfer, arguably the most straightforward transfer approach. In its simplest implementation, it only involves fine-tuning a pre-trained model. Accordingly, the employed network architectures are simple, none of the reviewed research papers used advanced DNN building blocks like Transformer, which are common in computer vision and language modeling. We expect this type of architecture with suitable modifications and/or pre-trained parameters to spread to more niche fields. Despite this, the survey suggests that deep transfer learning approaches have huge potential and promise for solving more complex and dynamic anomaly detection tasks in industry. As the field is still in an early stage, more R&D is expected to fully realize the potential of deep transfer learning in increasingly complex settings.

In the end, we highlight the importance of considering feasibility, reliability, explainability, and real-time data stream when designing a transfer learning system for time series anomaly detection. After carefully discussing open challenges, we gave practical directions for time series anomaly detection solution design and deep transfer learning implementation. In our view, the following directions hold the greatest potential for future work:

a: Automatic selection of transferable features [57]

It refers to methods for selecting and transferring only the relevant knowledge for the new tasks from the base model. This could involve the use of techniques such as selective fine-tuning and distillation to identify the most important features learned from source domains [30], [144].

b: Investing into advanced deep transfer learning schemes and DNN models

The conceptionally simplest parameter transfer approach has the advantage of being readily applicable by interdisciplinary teams without ML research experience. However, it seems promising to invest in testing more sophisticated deep transfer learning approaches according to different use cases, such as mapping transfer, adversarial transfer, etc. The same applies

to testing diverse DNN models besides straightforward ones. Recently, large models have been used in time series anomaly detection. For example, Xu *et al.* propose the Anomaly Transformer with a new anomaly-attention mechanism to compute the association discrepancy [145]. A minimax strategy is devised to amplify the normal-abnormal distinguishability of the association discrepancy. On the other hand, Pintilie *et al.* leverage diffusion models for multivariate time series anomaly detection [146]. They train two diffusion-based models that outperform strong transformer-based methods on synthetic datasets and are competitive on real-world data. Additionally, their DiffusionAE model is more robust to different levels and the number of anomaly types. These large models have proven to be effective and advantageous given certain data and tasks. It's important to note that their effectiveness also depends on the characteristics of the time series data and the requirements of the anomaly detection task. Additionally, model computational efficiency and interpretability should be considered, especially in real-time or resource-constrained industrial applications.

c: Data-centric approach to real-time anomaly detection

The data-centric approach focuses on improving ML models by ensuring high-quality labeled data [147] using techniques such as re-labeling, re-weighting, or data augmentation [148]. Currently, a human-in-the-loop solution is still needed. Frameworks have been proposed to assist annotators with graph-based algorithms such as nearest neighbor graphs [84], decision trees [149], or factor graphs [150]. Although these methods have proven to be effective, a more automated process is a goal for future research.

d: Leveraging generative AI

Generative models like GANs and diffusion models can generate synthetic time series data, making them valuable for data augmentation. Augmenting the original data with synthetic samples can enhance the deep learning models' robustness, especially in real-world applications where target data is limited. These models can also be leveraged to examine anomalies and generate anomalies to help alleviate the imbalance within the data [151].

e: Integration with other ML methods

To develop robust AI solutions for time series anomaly detection in the industry, relying solely on transfer learning is insufficient. Future strategies should integrate other ML approaches, including continuous learning, meta-learning, and federated learning.

LIST OF ACRONYMS

AI	Artificial Intelligence
CNN	Convolutional Neural Network
DAN	Deep Adaptation Networks
DNN	Deep Neural Network
FCL	Fully Connected layer

GAN	Generative Adversarial Network
LSTM	Long Short term memory
ML	Machine Learning
MMD	Maximum Mean Discrepancy
RNN	Recurrent Neural Network
SAE	Sparse Auto-Encoder

ACKNOWLEDGMENT

We would like to acknowledge Claudio Riginio for his constructive comments on an earlier draft of this paper and for his assistance with the illustrations.

REFERENCES

- [1] H. Kagermann, W.-D. Lukas, and W. Wahlster, "Industrie 4.0: Mit dem internet der dinge auf dem weg zur 4. industriellen revolution," *VDI Nachrichten*, vol. 13, no. 1, pp. 2–3, 2011.
- [2] V. Roblek, M. Meško, and A. Krapež, "A complex view of industry 4.0," *SAGE Open*, vol. 6, no. 2, Apr. 2016.
- [3] L. Wang, M. Törmgren, and M. Onori, "Current status and advancement of cyber-physical systems in manufacturing," *J. Manuf. Syst.*, vol. 37, pp. 517–527, Oct. 2015.
- [4] S. Jeschke, C. Brecher, T. Meisen, D. Özdemir, and T. Eschert, "Industrial internet of things and cyber manufacturing systems," in *Industrial Internet of Things: Cybermanufacturing Systems*. Springer, 2017, pp. 3–19.
- [5] L. S. Dalenogare, G. B. Benitez, N. F. Ayala, and A. G. Frank, "The expected contribution of Industry 4.0 technologies for industrial performance," *Int. J. Prod. Econ.*, vol. 204, pp. 383–394, Oct. 2018.
- [6] H. Kagermann, "Chancen von Industrie 4.0 nutzen," in *Handbuch Industrie 4.0 Bd.4: Allgemeine Grundlagen*. Berlin, Heidelberg: Springer, 2017, pp. 237–248.
- [7] S. M. A. A. Abir, A. Anwar, J. Choi, and A. S. M. Kayes, "IoT-Enabled smart energy grid: Applications and challenges," *IEEE Access*, vol. 9, pp. 50 961–50 981, 2021.
- [8] Y.-J. Park, S.-K. S. Fan, and C.-Y. Hsu, "A review on fault detection and process diagnostics in industrial processes," *Processes*, vol. 8, no. 9, p. 1123, Sep. 2020.
- [9] Y. Liao, I. Ragai, Z. Huang, and S. Kerner, "Manufacturing process monitoring using time-frequency representation and transfer learning of deep neural networks," *J. Manuf. Processes*, vol. 68, pp. 231–248, Aug. 2021.
- [10] Y. Lockner and C. Hopmann, "Induced network-based transfer learning in injection molding for process modelling and optimization with artificial neural networks," *IJAMT*, vol. 112, no. 11, pp. 3501–3513, Feb. 2021.
- [11] B. Maschler, T. Knodel, and M. Weyrich, "Towards deep industrial transfer learning for anomaly detection on time series data," in *Proc. 26th IEEE ETFA*, Sep. 2021, pp. 01–08.
- [12] T. Wen and R. Keyes, "Time series anomaly detection using convolutional neural networks and transfer learning," May 2019, arXiv:1905.13628.
- [13] Y. Xu, Y. Sun, X. Liu, and Y. Zheng, "A digital-twin-assisted fault diagnosis using deep transfer learning," *IEEE Access*, vol. 7, pp. 19 990–19 999, 2019.
- [14] W. Mao, D. Zhang, S. Tian, and J. Tang, "Robust detection of bearing early fault based on deep transfer learning," *Electronics*, vol. 9, no. 2, p. 323, Feb. 2020.
- [15] W. Wang *et al.*, "Anomaly detection of industrial control systems based on transfer learning," *Tsinghua Sci. Technol.*, vol. 26, no. 6, pp. 821–832, Dec. 2021.
- [16] M. Canizo, I. Triguero, A. Conde, and E. Onieva, "Multi-head CNN-RNN for multi-time series anomaly detection: An industrial case study," *Neurocomputing*, vol. 363, pp. 246–260, Oct. 2019.
- [17] M. Weber, C. Doblander, and P. Mandl, "Towards the detection of building occupancy with synthetic environmental data," Oct. 2020, arXiv:2010.04209.
- [18] A. N. Sayed, Y. Himeur, and F. Bensaali, "From time-series to 2D images for building occupancy prediction using deep transfer learning," *EAAI*, vol. 119, p. 105786, Mar. 2023.
- [19] Y. Yao, D. Ge, J. Yu, and M. Xie, "Model-based deep transfer learning method to fault detection and diagnosis in nuclear power plants," *Front. Energy Res.*, vol. 10, 2022.
- [20] M. Abdallah, W. J. Lee, N. Raghunathan, C. Mousoulis, J. W. Sutherland, and S. Bagchi, "Anomaly detection through transfer learning in agriculture and manufacturing IoT systems," Feb. 2021, arXiv:2102.05814.
- [21] C. Panjapornpon, S. Bardeeniz, M. A. Hussain, and P. Chomchai, "Explainable deep transfer learning for energy efficiency prediction based on uncertainty detection and identification," *Energy and AI*, vol. 12, p. 100224, Apr. 2023.
- [22] H. Dhillon and A. Haque, "Towards network traffic monitoring using deep transfer learning," in *Proc. IEEE 19th TrustCom*, Dec. 2020, pp. 1089–1096.
- [23] P. Xiong, Y. Zhu, Z. Sun, Z. Cao, M. Wang, and Y. Zheng, "Application of transfer learning in continuous time series for anomaly detection in commercial aircraft flight data," in *Proc. IEEE SmartCloud*, Sep. 2018, pp. 13–18.
- [24] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE TPAMI*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.
- [25] B. Maschler, H. Vietz, H. Tercan, C. Bitter, T. Meisen, and M. Weyrich, "Insights and example use cases on industrial transfer learning," *Procedia CIRP*, vol. 107, pp. 511–516, 2022.
- [26] B. Maschler and M. Weyrich, "Deep transfer learning for industrial automation: A review and discussion of new techniques for data-driven machine learning," *IEEE Ind. Electron. Mag.*, vol. 15, no. 2, pp. 65–75, 2021.
- [27] F. Zhuang *et al.*, "A comprehensive survey on transfer learning," *Proc. IEEE*, vol. 109, no. 1, pp. 43–76, Jan. 2021.
- [28] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
- [29] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *Proc. ICANN 2018*, 2018, pp. 270–279.
- [30] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. NeurIPS*, vol. 27, 2014.
- [31] F. Yu, X. Xiu, and Y. Li, "A survey on deep transfer learning and beyond," *Mathematics*, vol. 10, no. 19, p. 3619, Jan. 2022.
- [32] K. Choi, J. Yi, C. Park, and S. Yoon, "Deep learning for anomaly detection in time-series data: review, analysis, and guidelines," *IEEE Access*, vol. 9, pp. 120 043–120 065, 2021.
- [33] R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: A survey," Jan. 2019, arXiv:1901.03407.
- [34] L. Torrey and J. Shavlik, "Transfer learning," *Handbook of Research on Machine Learning Applications*, Jan. 2009.
- [35] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, 1997.
- [36] S. Ruder, "An overview of multi-task learning in deep neural networks," Jun. 2017, arXiv:1706.05098.
- [37] W. M. Kouw and M. Loog, "An introduction to domain adaptation and transfer learning," Jan. 2019, arXiv:1812.11806.
- [38] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, Jan. 2015.
- [39] Y. Lukic, C. Vogt, O. Dürr, and T. Stadelmann, "Speaker identification and clustering using convolutional neural networks," in *Proc. 26th IEEE MLSP*, Sep. 2016, pp. 1–6.
- [40] T. Stadelmann *et al.*, "Deep learning in the wild," in *Proc. 8th ANNPR*, 2018, pp. 17–38.
- [41] J. Schmidhuber, "Annotated history of modern AI and deep learning," Dec. 2022, arXiv:2212.11279.
- [42] Q.-Q. He, S. W. I. Siu, and Y.-W. Si, "Instance-based deep transfer learning with attention for stock movement prediction," *Appl. Intell.*, vol. 53, no. 6, pp. 6887–6908, Jul. 2022.
- [43] M. Amirian *et al.*, "PrepNet: A convolutional auto-encoder to homogenize CT scans for cross-dataset medical image analysis," in *Proc. 14th CISP-BMEI*, Oct. 2021, pp. 1–7.
- [44] T. Wang, J. Huan, and M. Zhu, "Instance-based deep transfer learning," in *Proc. IEEE WACV*, Jan. 2019, pp. 367–375.
- [45] R. Bommasani *et al.*, "On the opportunities and risks of foundation models," Jul. 2022, arXiv:2108.07258.
- [46] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT 2019*, Jun. 2019, pp. 4171–4186.

- [47] K. Zhang, S. Wang, S. Wang, and Q. Xu, "Anomaly detection of control moment gyroscope based on working condition classification and transfer learning," *Applied Sciences*, vol. 13, no. 7, p. 4259, Jan. 2023.
- [48] A. Vaswani *et al.*, "Attention is all you need," in *Proc. NeurIPS*, vol. 30, 2017.
- [49] Y. Dou, M. Forbes, R. Koncel-Kedziorski, N. A. Smith, and Y. Choi, "Is GPT-3 text indistinguishable from human text? Scarecrow: A framework for scrutinizing machine text," in *Proc. 60th Annu. Meeting. ACL*, May 2022, pp. 7250–7274.
- [50] N. Alexandr, O. Irina, K. Tatyana, K. Inessa, and P. Arina, "Fine-tuning GPT-3 for Russian text summarization," in *Proc. Data Science and Intelligent Systems*, 2021, pp. 748–757.
- [51] Y. Guo, H. Shi, A. Kumar, K. Grauman, T. Rosing, and R. Feris, "Spot-Tune: Transfer learning through adaptive fine-tuning," in *Proc. CVPR*, Jun. 2019, pp. 4800–4809.
- [52] P. Sager, S. Salzmann, F. Burn, and T. Stadelmann, "Unsupervised domain adaptation for vertebrae detection and identification in 3D CT volumes using a domain sanity loss," *J. Imaging*, vol. 8, no. 8, 2022.
- [53] T. Brown *et al.*, "Language models are few-shot learners," in *Proc. NeurIPS*, vol. 33, 2020, pp. 1877–1901.
- [54] Y. Wang, J. Yan, X. Ye, Q. Jing, J. Wang, and Y. Geng, "Few-shot transfer learning with attention mechanism for high-voltage circuit breaker fault diagnosis," *IEEE Trans. Ind. Appl.*, vol. 58, no. 3, pp. 3353–3360, May 2022.
- [55] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: Maximizing for domain invariance," Dec. 2014, arXiv:1412.3474.
- [56] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proc. ICML*, Jul. 2017, pp. 2208–2217.
- [57] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. ICML*, vol. 37, Jul. 2015, pp. 97–105.
- [58] X. Zhang, F. X. Yu, S.-F. Chang, and S. Wang, "Deep transfer network: Unsupervised domain adaptation," Mar. 2015, arXiv:1503.00591.
- [59] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *Proc. CVPR*, Jul. 2017, pp. 5385–5394.
- [60] E. Soleimani and E. Nazerfard, "Cross-subject transfer learning in human activity recognition systems using generative adversarial networks," *Neurocomputing*, vol. 426, pp. 26–34, Feb. 2021.
- [61] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko, "Simultaneous deep transfer across domains and tasks," in *Proc. ICCV*, Dec. 2015, pp. 4068–4076.
- [62] Y. Ozyurt, S. Feuerriegel, and C. Zhang, "Contrastive learning for unsupervised domain adaptation of time series," Feb. 2023, arXiv:2206.06243.
- [63] Y. Ganin *et al.*, "Domain-adversarial training of neural networks," in *Domain Adaptation in Computer Vision Applications*. Springer, 2017, pp. 189–209.
- [64] H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, and M. Marchand, "Domain-adversarial neural networks," Feb. 2015, arXiv:1412.4446.
- [65] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. CVPR*, Jul. 2017, pp. 7167–7176.
- [66] Z. Sun, M. Wang, and L. Li, "Multilingual translation via grafting pre-trained language models," in *Proc. EMNLP 2021*, Nov. 2021, pp. 2735–2747.
- [67] M. Glass *et al.*, "Span selection pre-training for question answering," in *Proc. 58th Annu. Meeting. ACL*, Jul. 2020, pp. 2773–2782.
- [68] L. Tugener, J. Schmidhuber, and T. Stadelmann, "Is it enough to optimize CNN architectures on ImageNet?" *Frontiers in Computer Science*, vol. 4, 2022.
- [69] J. Shen, Y. Qu, W. Zhang, and Y. Yu, "Wasserstein distance guided representation learning for domain adaptation," in *Proc. AAAI*, Feb. 2018, pp. 4058–4065.
- [70] W. Dai, G.-R. Xue, Q. Yang, and Y. Yu, "Co-clustering based classification for out-of-domain documents," in *Proc. 13th ACM SIGKDD*, ser. KDD '07, Aug. 2007, pp. 210–219.
- [71] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *Proc. BMVC 2014*, 2014, pp. 6.1–6.12.
- [72] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. NeurIPS*, vol. 27, 2014.
- [73] J. Schmidhuber, "Generative Adversarial Networks are special cases of Artificial Curiosity (1990) and also closely related to Predictability Minimization (1991)," *Neural Networks*, vol. 127, pp. 58–66, 2020.
- [74] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural Networks*, vol. 113, pp. 54–71, May 2019.
- [75] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Generalizing from a few examples: A survey on few-shot learning," *ACM Comput. Surv.*, vol. 53, no. 3, pp. 63:1–63:34, Jun. 2020.
- [76] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE TPAMI*, vol. 28, no. 4, pp. 594–611, Apr. 2006.
- [77] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *Proc. CVPR*, Jun. 2009, pp. 951–958.
- [78] K. Zhou, Z. Liu, Y. Qiao, T. Xiang, and C. C. Loy, "Domain generalization: A survey," *IEEE TPAMI*, vol. 45, no. 4, pp. 4396–4415, Apr. 2023.
- [79] G. Blanchard, G. Lee, and C. Scott, "Generalizing from several related classification tasks to a new unlabeled sample," in *Proc. NeurIPS*, vol. 24, 2011.
- [80] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. 34th ICML*. PMLR, Jul. 2017, pp. 1126–1135.
- [81] T. Hospedales, A. Antoniou, P. Micaelli, and A. Storkey, "Meta-learning in neural networks: A survey," *IEEE TPAMI*, vol. 44, no. 09, pp. 5149–5169, Sep. 2022.
- [82] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *Int. J. Comput. Vision*, vol. 129, no. 6, pp. 1789–1819, Jun. 2021.
- [83] X. Liu *et al.*, "Self-supervised learning: Generative or contrastive," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 1, pp. 857–876, Jan. 2023.
- [84] H. Bai, M. Cao, P. Huang, and J. Shan, "Self-supervised semi-supervised learning for data labeling and quality evaluation," in *NeuIPS. Workshop*, Nov. 2021.
- [85] H. Tercan, A. Guajardo, J. Heinisch, T. Thiele, C. Hopmann, and T. Meisen, "Transfer-learning: Bridging the gap between real and simulation data for machine learning in injection molding," *Procedia CIRP*, vol. 72, pp. 185–190, Jan. 2018.
- [86] N. Göritz, M. Kloft, K. Rieck, and U. Brefeld, "Toward supervised anomaly detection," *JAIR*, vol. 46, pp. 235–262, Feb. 2013.
- [87] Y. Zhang, Y. Chen, J. Wang, and Z. Pan, "Unsupervised deep anomaly detection for multi-sensor time-series signals," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 2, pp. 2118–2132, Feb. 2023.
- [88] T. Stadelmann, V. Volkachev, B. Sick, J. Stampfli, and O. Dürr, "Beyond ImageNet: Deep learning in industrial practice," in *Applied Data Science: Lessons Learned for the Data-Driven Business*. Springer, 2019, pp. 205–232.
- [89] D. M. Hawkins, *Identification of outliers*. Dordrecht: Springer Netherlands, 1980, vol. 11.
- [90] H. Zhu, C. Yi, S. Rho, S. Liu, and F. Jiang, "An interpretable multivariate time-series anomaly detection method in cyber-physical systems based on adaptive mask," *IEEE Internet Things J.*, pp. 1–1, 2023.
- [91] S. Schmidl, P. Wenig, and T. Papenbrock, "Anomaly detection in time series: A comprehensive evaluation," *Proc. VLDB Endow.*, vol. 15, no. 9, pp. 1779–1797, May 2022.
- [92] J. Audibert, P. Michiardi, F. Guyard, S. Marti, and M. A. Zuluaga, "USAD: Unsupervised anomaly detection on multivariate time series," in *Proc. 26th ACM SIGKDD*, ser. KDD '20, 2020, pp. 3395–3404.
- [93] P. Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, and G. Shroff, "LSTM-based encoder-decoder for multi-sensor anomaly detection," Jul. 2016, arXiv:1607.00148.
- [94] Y. Wei, J. Jang-Jaccard, W. Xu, F. Sabrina, S. Camtepe, and M. Boulic, "LSTM-autoencoder-based anomaly detection for indoor air quality time-series data," *IEEE Sens. J.*, vol. 23, no. 4, pp. 3738–3800, Feb. 2023.
- [95] F. Zeng, M. Chen, C. Qian, Y. Wang, Y. Zhou, and W. Tang, "Multivariate time series anomaly detection with adversarial transformer architecture in the Internet of Things," *Future Gener. Comput. Syst.*, vol. 144, pp. 244–255, Jul. 2023.
- [96] D. Li, D. Chen, B. Jin, L. Shi, J. Goh, and S.-K. Ng, "MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks," in *Proc. ICANN 2019*, 2019, pp. 703–716.
- [97] Z. Niu, K. Yu, and X. Wu, "LSTM-based VAE-GAN for time-series anomaly detection," *Sensors*, vol. 20, no. 13, p. 3738, Jan. 2020.
- [98] M. A. Bashar and R. Nayak, "TanoGAN: Time series anomaly detection with generative adversarial networks," in *Proc. IEEE SSCI*, Dec. 2020, pp. 1778–1785.
- [99] J. Kim, H. Kang, and P. Kang, "Time-series anomaly detection with stacked Transformer representations and 1D convolutional network," *EAAI*, vol. 120, p. 105964, Apr. 2023.

- [100] A. Deng and B. Hooi, "Graph neural network-based anomaly detection in multivariate time series," *Proc. AAAI*, vol. 35, no. 5, pp. 4027–4035, May 2021.
- [101] C. Tang, L. Xu, B. Yang, Y. Tang, and D. Zhao, "GRU-based interpretable multivariate time series anomaly detection in industrial control system," *Comput. Secur.*, vol. 127, p. 103094, Apr. 2023.
- [102] C. Ding, S. Sun, and J. Zhao, "MST-GAT: A multimodal spatial-temporal graph attention network for time series anomaly detection," *Inf. Fusion*, vol. 89, pp. 527–536, Jan. 2023.
- [103] Y. Himeur, A. Alsalemi, F. Bensaali, and A. Amira, "A novel approach for detecting anomalous energy consumption based on micro-moments and deep neural networks," *Cognit. Comput.*, vol. 12, no. 6, pp. 1381–1401, Nov. 2020.
- [104] Y. Yang, C. Zhang, T. Zhou, Q. Wen, and L. Sun, "DCdetector: Dual attention contrastive representation learning for time series anomaly detection," in *Proc. 29th ACM SIGKDD*, ser. KDD '23, Aug. 2023, pp. 3033–3045.
- [105] W. Li *et al.*, "A perspective survey on deep transfer learning for fault diagnosis in industrial scenarios: Theories, applications and challenges," *Mech. Syst. Signal Process.*, vol. 167, p. 108487, Mar. 2022.
- [106] J. Ma, J. C. Cheng, C. Lin, Y. Tan, and J. Zhang, "Improving air quality prediction accuracy at larger temporal resolutions using deep learning and transfer learning techniques," *Atmos. Environ.*, vol. 214, p. 116885, Oct. 2019.
- [107] I. Rosenberg, G. Sicard, and E. David, "End-to-end deep neural networks and transfer learning for automatic analysis of nation-state malware," *Entropy*, vol. 20, no. 5, p. 390, May 2018.
- [108] Q. Pan, Y. Bao, and H. Li, "Transfer learning-based data anomaly detection for structural health monitoring," *Struct. Health Monit.*, vol. 22, no. 5, pp. 3077–3091, Jan. 2023.
- [109] G. Li, L. Chen, J. Liu, and X. Fang, "Comparative study on deep transfer learning strategies for cross-system and cross-operation-condition building energy systems fault diagnosis," *Energy*, vol. 263, p. 125943, Jan. 2023.
- [110] O. Serradilla, E. Zugasti, J. Ramirez de Okariz, J. Rodriguez, and U. Zurutuza, "Adaptable and explainable predictive maintenance: Semi-supervised deep learning for anomaly detection and diagnosis in press machine data," *Applied Sciences*, vol. 11, no. 16, p. 7376, Jan. 2021.
- [111] J. Zraggen, M. Ulmer, E. Jarlskog, G. Pizza, and L. G. Huber, "Transfer learning approaches for wind turbine fault detection using deep learning," *PHME 2021*, vol. 6, no. 1, pp. 12–12, Jun. 2021.
- [112] M. Zabin, H.-J. Choi, and J. Uddin, "Hybrid deep transfer learning architecture for industrial fault diagnosis using Hilbert transform and DCNN-LSTM," *J. Supercomput.*, vol. 79, no. 5, pp. 5181–5200, Mar. 2023.
- [113] H. Tercan, A. Guajardo, and T. Meisen, "Industrial transfer learning: Boosting machine learning in production," in *Proc. IEEE 17th INDIN*, vol. 1, Jul. 2019, pp. 274–279.
- [114] Y. Lockner, C. Hopmann, and W. Zhao, "Transfer learning with artificial neural networks between injection molding processes and different polymer materials," *J. Manuf. Processes*, vol. 73, pp. 395–408, Jan. 2022.
- [115] S. Gellrich *et al.*, "Deep transfer learning for improved product quality prediction: A case study of Aluminum gravity die casting," *Procedia CIRP*, vol. 104, pp. 912–917, Jan. 2021.
- [116] M. Abdallah *et al.*, "Anomaly detection and inter-sensor transfer learning on smart manufacturing datasets," *Sensors*, vol. 23, no. 1, p. 486, Jan. 2023.
- [117] B. Maschler, T. T. Huong Pham, and M. Weyrich, "Regularization-based continual learning for anomaly detection in discrete manufacturing," *Procedia CIRP*, vol. 104, pp. 452–457, Jan. 2021.
- [118] J. Park, B. Kim, and H. Kim, "MENDEL: Time series anomaly detection using transfer learning for industrial control systems," in *Proc. IEEE BigComp*, Feb. 2023, pp. 261–268.
- [119] P. Liang, H.-D. Yang, W.-S. Chen, S.-Y. Xiao, and Z.-Z. Lan, "Transfer learning for aluminium extrusion electricity consumption anomaly detection via deep neural networks," *Int. J. Comput. Integr. Manuf.*, vol. 31, no. 4–5, pp. 396–405, Apr. 2018.
- [120] A. Copiaco *et al.*, "An innovative deep anomaly detection of building energy consumption using energy time-series images," *EAAI*, vol. 119, p. 105775, Mar. 2023.
- [121] C. Xu, J. Wang, J. Zhang, and X. Li, "Anomaly detection of power consumption in yarn spinning using transfer learning," *Comput. Ind. Eng.*, vol. 152, p. 107015, Feb. 2021.
- [122] F. D. Simone and F. Amigoni, "Analysis of machine learning methods for anomaly detection of power consumption in buildings," Master's thesis, Politecnico di Milano, 2021, available: <https://www.politesi.polimi.it/handle/10589/183344>.
- [123] R.-J. Hsieh, J. Chou, and C.-H. Ho, "Unsupervised online anomaly detection on multivariate sensing time series data for smart manufacturing," in *Proc. IEEE 12th SOCA*, Nov. 2019, pp. 90–97.
- [124] O. Serradilla, E. Zugasti, J. Rodriguez, and U. Zurutuza, "Deep learning models for predictive maintenance: a survey, comparison, challenges and prospects," *Appl. Intell.*, vol. 52, no. 10, pp. 10934–10964, Aug. 2022.
- [125] L. Wen, L. Gao, and X. Li, "A new deep transfer learning based on sparse auto-encoder for fault diagnosis," *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 49, no. 1, pp. 136–144, Jan. 2019.
- [126] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. MICCAI*, 2015, pp. 234–241.
- [127] C. Sun and others, "Deep transfer learning based on sparse autoencoder for remaining useful life prediction of tool in manufacturing," *IEEE Trans. Ind. Inf.*, vol. 15, no. 4, pp. 2416–2425, Apr. 2019.
- [128] A. Copiaco, Y. Himeur, A. Amira, W. Mansoor, F. Fadli, and S. Atalla, "Exploring deep time-series imaging for anomaly detection of building energy consumption," in *Proc. IEEE CSDE*, Dec. 2022, pp. 1–5.
- [129] A. N. Sayed, Y. Himeur, and F. Bensaali, "Deep and transfer learning for building occupancy detection: A review and comparative analysis," *EAAI*, vol. 115, p. 105254, Oct. 2022.
- [130] M. Goldstein and S. Uchida, "A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data," *PLoS one*, vol. 11, no. 4, pp. 1–31, Apr. 2016.
- [131] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Networks*, vol. 106, pp. 249–259, Oct. 2018.
- [132] K. A. Smith-Jentsch, E. Salas, and M. T. Brannick, "To transfer or not to transfer? Investigating the combined effects of trainee characteristics, team leader support, and team climate," *J. Appl. Psychol.*, vol. 86, pp. 279–92, May 2001.
- [133] X. Glorot, A. Bordes, and Y. Bengio, "Domain adaptation for large-scale sentiment classification: a deep learning approach," in *Proc. 28th ICML*, Jun. 2011, pp. 513–520.
- [134] W. Jiang, Y. Hong, B. Zhou, X. He, and C. Cheng, "A GAN-based anomaly detection approach for imbalanced industrial time series," *IEEE Access*, vol. 7, pp. 143 608–143 619, 2019.
- [135] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. ICLR*, 2015.
- [136] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *JAIR*, vol. 16, pp. 321–357, Jun. 2002.
- [137] M. F. Ijaz, G. Alfian, M. Syafrudin, and J. Rhee, "Hybrid prediction model for type 2 diabetes and hypertension using DBSCAN-based outlier detection, synthetic minority over sampling technique (SMOTE), and random forest," *Applied Sciences*, vol. 8, no. 8, p. 1325, Aug. 2018.
- [138] S. Mokhtari, A. Abbaspour, K. K. Yen, and A. Sargolzaei, "A machine learning approach for anomaly detection in industrial control systems based on measurement data," *Electronics*, vol. 10, no. 4, p. 407, Jan. 2021.
- [139] A. Kirillov *et al.*, "Segment anything," in *Proc. ICCV*, 2023, pp. 4015–4026.
- [140] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proc. CVPR*, Jun. 2022, pp. 10 684–10 695.
- [141] A. Vrontzos, B. Kainz, and C. M. Gilligan-Lee, "Estimating categorical counterfactuals via deep twin networks," *Nat. Mach. Intell.*, vol. 5, no. 2, pp. 159–168, Feb. 2023.
- [142] S. Wang *et al.*, "When edge meets learning: Adaptive control for resource-constrained distributed machine learning," in *Proc. IEEE INFOCOM 2018*, Apr. 2018, pp. 63–71.
- [143] M. M. Amiri and D. Gündüz, "Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air," in *Proc. IEEE ISIT*, Jul. 2019, pp. 1432–1436.
- [144] W. Ge and Y. Yu, "Borrowing treasures from the wealthy: Deep transfer learning through selective joint fine-tuning," in *Proc. IEEE CVPR*, Jul. 2017, pp. 10–19.
- [145] J. Xu, H. Wu, J. Wang, and M. Long, "Anomaly Transformer: Time series anomaly detection with association discrepancy," in *ICLR*, Oct. 2021.
- [146] I. Pintilie, A. Manolache, and F. Brad, "Time series anomaly detection using diffusion-based models," Nov. 2023, arXiv:2311.01452.

- [147] T. Stadelmann, T. Klamt, and P. H. Merkt, "Data centrism and the core of Data Science as a scientific discipline," *Archives of Data Science, Series A*, vol. 8, no. 2, Mar. 2022.
- [148] P.-P. Luley, J. M. Deriu, P. Yan, G. A. Schatte, and T. Stadelmann, "From concept to implementation: the data-centric development process for AI in industry," in *Proc. 10th IEEE SDS*, 2023.
- [149] Z. Y.-C. Liu, S. Roychowdhury, S. Tarlow, A. Nair, S. Badhe, and T. Shah, "AutoDC: Automated data-centric processing," in *NeuIPS. Workshop*, Nov. 2021, available: <https://nips.cc/virtual/2021/38244>.
- [150] D. Kang, N. Arechiga, S. Pillai, P. D. Bailis, and M. Zaharia, "Finding label and model errors in perception data with learned observation assertions," in *Proc. ACM SIGMOD*, Jun. 2022, pp. 496–505.
- [151] M. Salem, S. Taheri, and J. S. Yuan, "Anomaly generation using generative adversarial networks in host-based intrusion detection," in *Proc. IEEE UEMCON*, Nov. 2018, pp. 683–687.



currently working on applying deep learning to industrial applications.

PENG YAN received M.Sc degrees in Informatics from University of Zurich, Zurich, Switzerland, in 2022. He is currently a PhD student in Data Science at the University of Zurich and a research assistant at ZHAW School of Engineering in Winterthur, Switzerland. He is a member of the Centre for Artificial Intelligence and the Machine Vision, Perception and Cognition Group. His research interests include deep learning, time series analysis, computer vision and intelligent algorithms. He is



researching application-oriented methods for industrial and medical domains.

AHMED ABDULKADIR is a senior research scientist at the Center of Artificial Intelligence of the ZHAW School of Engineering in Winterthur, Switzerland. He studied Life Sciences at the École Polytechnique Fédérale de Lausanne, Switzerland, and received his Doctor of Engineering degree from the Albert-Ludwigs-Universität, Freiburg-im-Breisgau, Germany. Before his appointment, Ahmed researched machine learning for medical and biological applications. He is currently re-



models, ensuring their reliability and effectiveness in practical applications.

PAUL-PHILIPP LULEY is a research scientist at Julius Bär, specializing in robust deep learning techniques with a focus on MLOps. He earned a B.Sc. degree in Biomedical Engineering from the UAS Technikum Wien and is currently pursuing an M.Sc. degree in Computer Science at the University of Zurich. His research interests lie at the intersection of robust deep learning and MLOps, where he aims to develop cutting-edge solutions for operationalizing and maintaining machine learning



video, and embedded computing.
VOLUME 11, 2023

MATTHIAS ROSENTHAL is a professor for Multiprocessor Systems at the ZHAW Zurich University of Applied Sciences. He is also the Head of Research/Focus Area, Realtime Platforms. He received a Ph.D degree from ETH Zürich, Switzerland, in 1997. He has been a senior lecturer and head of research area at ZHAW since 2014. His research interests encompass a wide array of areas, including Multiprocessors, Hybrid multicore systems, distributed digital signal processing, audio,



with a large variety of differently conditioned data depending on the application. As team lead business development/project management at the Kistler Innovation Lab, he oversees multiple projects to develop Kistler's data-based digital services and the corresponding business.

GERRIT A. SCHATTE is a mechanical engineer with a passion for process measuring and control technology. He received his Ph.D from TUM, experimentally investigating heat transfer to supercritical fluids. Following his doctorate, his interest in measuring systems led him to join Kistler, the world market leader for dynamic mechanical measuring systems. Throughout his roles in research and industry, he has worked with quantitative methods in both R&D and production settings



logical networks and then to reverse-engineer their functionality as logical, reproducible algorithms.

BENJAMIN F. GREWE is a professor at the Institute of Neuroinformatics at the University of Zurich and ETH Zurich. He received his Ph.D degree in Neuroinformatics from ETH Zurich, Switzerland, in 2010, and then worked as a Postdoc in Systems Neuroscience at Stanford University, USA from 2011 to 2016. His research focuses on the intersection of biological and artificial intelligence. His long-term vision is to extract fundamental principles of network learning from real biological



industry for several years prior to his appointment at the ZHAW. His current research focuses on robust deep learning to solve diverse pattern recognition tasks such as document analysis or industrial or medical computer vision.

THILO STADELMANN is professor of AI/ML at the ZHAW School of Engineering in Winterthur, Switzerland, director of the ZHAW Centre for Artificial Intelligence and head of its Machine Perception and Cognition Group. He studied computer science in Giessen and Marburg and received his Doctor of Science degree from Marburg University, Germany, in 2010, where he worked on multimedia analysis and voice recognition. Thilo held engineering and leadership roles in the automotive

...

Originally published in TISMIR 7(1), 1-14, 2024. DOI 10.5334/tismir.157.

RESEARCH ARTICLE

Real World Music Object Recognition

Lukas Tuggener^{*†}, Raphael Emberger^{*†}, Adhiraj Ghosh^{*†}, Pascal Sager^{*†}, Yvan Putra Satyawan[‡], Javier Montoya[‡], Simon Goldschagg[§], Florian Seibold[§], Urs Gut[¶], Philipp Ackermann^{||}, Jürgen Schmidhuber^{**††}, and Thilo Stadelmann^{† ‡‡}

Abstract

We present solutions to two of the most pressing issues in contemporary optical music recognition (OMR). We improve recognition accuracy on low-quality, real-world (i.e. containing ageing, lighting, or dirt artefacts among others) input data and provide confidence-rated model outputs to enable efficient human post-processing. Specifically, we present (i) a sophisticated input augmentation scheme that can reduce the gap between sanitised benchmarks and realistic tasks through a combination of synthetic data and noisy perturbations of real-world documents; (ii) an adversarial discriminative domain adaptation method that can be employed to improve the performance of OMR systems on low-quality data; (iii) a combination of model ensembles and prediction fusion, which generates trustworthy confidence ratings for each prediction. We evaluate our contributions on a newly created test set consisting of manually annotated pages of varying real-world quality, sourced from International Music Score Library Project (*IMSLP*) / the Petrucci Music Library. With the presented data augmentation scheme, we achieve a doubling in detection performance from 36.0% to 73.3% on noisy real-world data compared to state-of-the-art training. This result is then combined with robust confidence ratings paving the way for OMR to be deployed in the real world. Additionally, we show the merits of unsupervised adversarial domain adaptation for OMR raising the 36.0% baseline to 48.9%.

All our code and data are freely available at: https://github.com/raember/s2anet/tree/TISMIR_publication.

Keywords: Optical Music Recognition, Deep Learning, Data Augmentation, Adversarial Training, Model Ensembles, Open Data

1. Introduction

Optical music recognition (OMR) (Rebelo et al., 2012; Calvo-Zaragoza et al., 2020) is a classical and challenging area of document analysis, that aims to convert images of written music to machine-readable, encoded form. A crucial component of any OMR pipeline is a music object recognition (MOR) system. In recent years MOR systems have reached greatly increased performance thanks to the adoption of deep

learning (Pacha et al., 2018b; Tuggener et al., 2018b) and the availability of large datasets (Hajič and Pecina, 2017; Tuggener et al., 2018a, 2021).

Despite these advancements, we have identified two major roadblocks that hold current MOR systems back from reaching their full potential in a practical, real-world setting. Even though deep neural networks have been consistently revolutionising different computer vision tasks like classification, object detection, segmentation, image retrieval, and many more, they often fail to replicate the benchmark performances and results on new domains. This issue is attributed to *cross-domain mismatch*, some of the problems surrounding this issue are highlighted below.

Firstly, the currently available (MOR) training datasets are either synthetically generated or scans of very high quality, which are visually very close to synthetic imagery. This causes the resulting

*These authors contributed equally

†Centre for Artificial Intelligence, ZHAW School of Engineering, Zurich University of Applied Sciences, Winterthur, Switzerland

‡Work done while with the ZHAW School of Engineering

§ScorePad AG, Erlenbach, Switzerland

¶Work done while studying at the ZHAW School of Engineering

|| Inst. of appl. Inform. Technology, ZHAW School of Engineering

**The Swiss AI Lab IDSIA, Lugano, Switzerland

†† AI Initiative, KAUST, Thuwal, Saudi Arabia

‡‡ Fellow, ECLT European Centre for Living Technology, Venice, Italy

1 INTRODUCTION

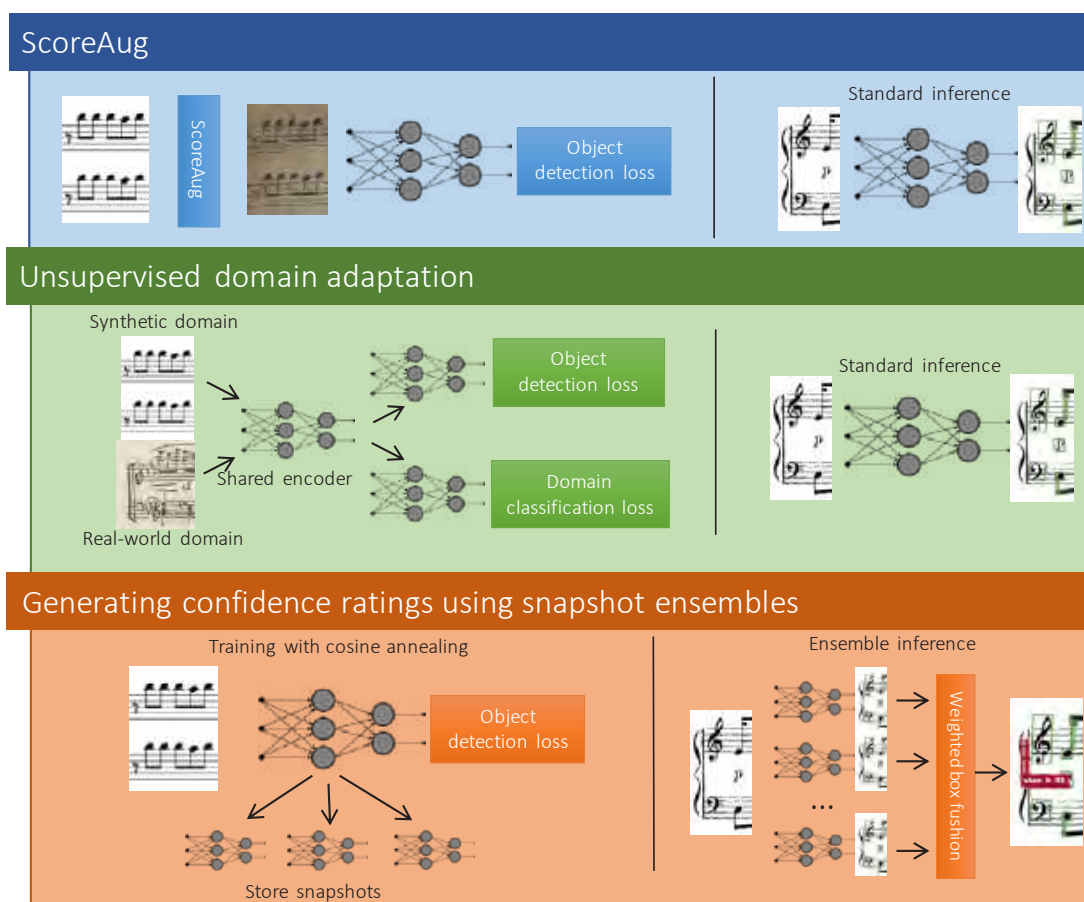


Figure 1: Graphical overview of the methods we contribute. *ScoreAug* (Section 5.1) in the top row, unsupervised domain adaptation (Section 5.2) in the center and snapshot-ensemble-based confidence ratings (Section 5.3) at the bottom.

detectors to perform very well on clean samples (Elezi et al., 2018; Pacha and Calvo-Zaragoza, 2018), but they struggle significantly when confronted with sub-optimal data quality as it is common in real-world applications (later referred to as real-world data). This can be scans of old or degraded pages, or for example, smartphone pictures under non-ideal conditions.

Secondly, deep neural networks are notoriously overconfident in their predictions – especially if an input lies outside the previously observed training data (Nguyen et al., 2015). This has great implications for the practical usability of MOR because it forces quality control, which is typically performed manually by humans, to check every detection with high diligence. This is particularly cumbersome for sheet music, where clusters of many tightly packed symbols are very common.

We attempt to resolve the issue of creating effective MOR systems for real-world musical sheet images by casting it as a *domain shift* problem. In order to bridge this domain gap between synthetic and realistic images, we propose two

approaches: *ScoreAug* (Section 5.1), which creates augmentations for diversity in feature distributions during training and unsupervised domain adaptation (UDA) (Section 5.2): Training on one data distribution enables the model to also perform well on different target distributions.

ScoreAug uses real-world, scanned blank pages with natural signs of degradation and combines them with the synthetic input from our initial dataset. By mending those two together, realistic-looking samples can be created on-the-fly. The result of that operation is that the model generalises more to real-world data, thereby bridging the domain shift.

Unsupervised domain adaptation addresses the cross-domain mismatch issue by manipulating the target domain samples (in our experiments the real-world images). In our research, we select a *domain-adversarial loss* in order to enforce target (real-world data) embeddings to be similar or close to the source (synthetic data) embeddings in a latent feature space. We bridge the gap between domains without the need for the generative modelling

2 SURVEY OF RELATED WORK

capabilities of adversarial models but with the help of a binary domain discriminator.

We address the problem of overconfident predictions by using an ensemble method. Good ensembles result when the predictions of the ensemble members are both accurate and have independent errors (Wen et al., 2020). Thus, the prediction confidence can be estimated over the predictions of several members and hence be better quantified.

We use SnapshotEnsemble (Huang et al., 2017), a method that creates ensemble members at no additional training cost. At inference time, each ensemble member makes a prediction independent of the other members. These predictions are fused into an average prediction with higher accuracy and more reliable confidence ratings and thus facilitate subsequent quality control.

In summary, the contributions of this paper are as follows (c.f. Figure 1):

- A MOR test dataset (*RealScores*, see Section 4) that contains 14 pages of real-world sheet music. The annotations for this data were newly created by hand and follow the class definitions and data structure of *DeepScoresV2* (Tuggener et al., 2021);
- *ScoreAug* (see Section 5.1): A sophisticated data augmentation scheme and training schedule using a combination of synthetic data (*DeepScoresV2*) and perturbations sourced from a diverse array of real-world documents (*IMSLP*), which are combined using randomised heuristics;
- An adversarial discriminative method for implementing unsupervised domain adaptation (see Section 5.2) in MOR for finding indiscriminate representations for the distributions of synthetic (*DeepScoresV2*) and real-world (*IMSLP*) features and bridging the gap between these domains in a latent feature space;
- Trustworthy confidence ratings (Section 5.3) for symbol level detections based on a prediction fusion algorithm that utilises confidence scores of ensemble outputs to calculate average predictions and confidence ratings.

The rest of this paper is organised as follows: Section 2 gives a thorough introduction to making OMR more robust in practice by surveying related work. In Section 3 we introduce our baseline model on which all of our solutions are built. Section 4 presents the *RealScores* dataset, a newly sourced and annotated small test set for real-world MOR. In Section 5 we present our proposed methods. Section 6 contains descriptions and results of all our experiments. Lastly, in Section 7 we draw conclusions and discuss possible future work.

2. Survey of Related Work

Music Object Detection Traditionally, OMR systems consisted of a cascade of components such as staff-line removal (Fujinaga, 2004; Dalitz et al., 2008), symbol segmentation (Bellini et al., 2001) and symbol classification (Toyama et al., 2006), which were built using classical computer vision methods. With the advent of increased computing power and the availability of large-scale datasets (Hajič and Pecina, 2017; Tuggener et al., 2021), deep-learning-based approaches (Schmidhuber, 2015) started to take over. Deep learning methods resulted in greatly increased performances in the above-mentioned tasks (Gallego and Calvo-Zaragoza, 2017). More recent works apply convolutional neural networks directly to the raw input data, making multi-step designs obsolete (Pacha et al., 2018a; Hajic Jr et al., 2018; Tuggener et al., 2018b). There are efforts to solve the whole OMR problem in one single step, as it is state of the art in related fields such as text (Chowdhury and Vig, 2018) or speech (Chiu et al., 2018) recognition. However, due to the high complexity of music notation, all existing solutions focus on a simplified problem such as mensural notation (Pugin, 2006) or monophonic scores, both typeset (van der Wel and Ullrich, 2017; Calvo-Zaragoza and Rizo, 2018b) and handwritten (Baró et al., 2018).

Input Data Augmentation Input data augmentation has a rich history in deep learning. However, it is mostly used to improve performance on a single domain. Typically, data augmentation consists of scaling, translations, and rotations Ciregan et al. (2012); Sato et al. (2015). On larger natural datasets such as ImageNet more sophisticated transforms like random cropping, image flipping, and colour normalization have become commonplace Krizhevsky et al. (2017). Generative adversarial networks have been employed to generate additional realistic training data Zhu et al. (2017). Recently, automatic search of optimal augmentation strategies on a per dataset basis has become the standard Cubuk et al. (2019).

There has already been some effort to address the domain gap between existing datasets and real-world data using data augmentation in the context of MOR. Datasets that have been altered to mimic realistic data have been created, either by applying a sequence of graphics filters (Calvo-Zaragoza and Rizo, 2018a) or by printing and scanning the data (Elezi et al., 2018).

To the best of our knowledge, this is the first work to present an input augmentation technique, that combines algorithmic distortions with real-world perturbations for MOR.

Domain Adaptation for Object Detection UDA is an unsupervised learning approach (Simmler et al., 2021) to transfer knowledge obtained from a source

4 THE DATA

domain with labelled data to a target domain with unlabelled data. One of the fundamental approaches in UDA was proposed by Tzeng et al. (2017), creating a generalised framework for adversarial adaptation in image classification.

Recently, UDA methods for tasks outside classification, such as object detection, have attracted increasing attention, which is the primary focus of our OMR models. Chen et al. (2018) was one of the pioneering works on this task. The authors observed image and instance level shifts and proposed segregated components to alleviate the domain discrepancy.

Adversarial approaches for discriminative UDA have recently reflected strong results in object detection (Zhu et al., 2019; Lehner et al., 2022; Li et al., 2022). The primary goal of most of the adversarial approaches addressed above is adversarial feature alignment between the source and target domain.

In the context of MOR, Mateiu et al. (2019) employed a domain adversarial neural network (Ganin and Lempitsky, 2015) to enable the classification of individual handwritten symbols in old music manuscripts. Castellanos et al. (2021) use UDA to improve document analysis (splitting of the input in a layered version containing different information, e.g. staves, notes or background) on historical music sheets.

To the best of our knowledge, this is the first work to employ and systematically evaluate UDA techniques for a full-fledged MOR system.

Confidence Ratings Most state-of-the-art approaches to estimate predictive uncertainty rely on ensembles (Gustafsson et al., 2020; Wen et al., 2020; von Oswald et al., 2021; Wenzel et al., 2020; Durasov et al., 2021; Xia et al., 2021; Huang et al., 2017). Bayesian deep learning approaches like MC-dropout have interesting properties but fail to deliver in practice due to computational or technical constraints (Dürr et al., 2020).

Since we train our models for 1000+ epochs and the input images are large (i.e. require a lot of memory), we focus on approaches known as “economic ensembles”, such as HypernetEnsembles von Oswald et al. (2021) or Masksembles Durasov et al. (2021). For these methods, the computational and memory costs do not increase linearly with the number of ensemble members and thus scale well with large deep learning models.

SnapshotEnsemble was proposed by Huang et al. (2017) and tries to achieve the seemingly paradoxical goal of producing an ensemble at no additional training cost. Their method leverages work on cyclic learning rate schedules (Smith, 2017). They lower the learning rate at a very fast pace, thus encouraging the model to converge quickly to its first local minimum.

DeepScoresV2 dataset	
Model	AP (overlap = 0.50)
Baseline model	89.3%
DWD	50.3%
Faster R-CNN	79.9%

Table 1: The AP at 0.5 overlap for our baseline model and two state-of-the-art models (DWD, Faster R-CNN Tugener et al. (2021)) on *DeepScoresV2*.

Then the optimisation is continued with a higher learning rate to dislodge the model from this local minimum again. This procedure is repeated multiple times. At each local minimum, the model is saved (i.e. a snapshot is taken). Ensembling the snapshots result in consistently lower error rates than single models. In this work, we exclusively employ SnapshotEnsembles due to their minimal compute requirements.

To the best of our knowledge, this is the first work to employ uncertainty measures in the context of OMR and systematically evaluate their merits.

3. Baseline Model

All our experiments are based on the *S²A-Net* architecture (Han et al., 2021), which allows for oriented detections unlike earlier methods (Tugener et al., 2018b; Pacha et al., 2018b). The *S²A-Net* is an anchor-based object detector that uses a single-shot alignment network to generate accurately oriented object detections. Its novel feature alignment and oriented detection modules are fed using a ResNet-based backbone (He et al., 2016) and feature-pyramid networks (Lin et al., 2017).

We achieve good results on the “oriented mode” of *DeepScoresV2* (Tugener et al., 2021) when training *S²A-Nets* by scaling the data with a factor of 0.5 and then using random crops of 1000 by 1000 pixels. We are able to conserve GPU memory whilst keeping high precision by just using a singular anchor ratio of 1.0 and a singular anchor scale of 4. We train our models with SGD using a learning rate of $\alpha = 2.5 \cdot 10^{-3}$ and a momentum of 0.9. Table 1 contains the average precision (AP) of our baseline model against two state-of-the-art models, which illustrates the competitive performance of our new *S²A-Net* based approach. The complete training details can be found in the published code.

4. The RealScores Data

So far, no real-world test data is available to benchmark models on. Such data is crucial to observe how well our models will perform when facing a domain gap. To create a benchmark dataset for real-world OMR, we sourced digitised music scores from the International Music Score Library Project

5 METHODS



Figure 2: Example snippets from two *RealScores* pages with ground truth annotations overlaid.

(*IMSLP*) / Petrucci Music Library¹. Of the downloaded music scores, only those with specific characteristics were considered for the new test set: The sheets had to be scans or photographs of music scores and be visibly non-synthetic, meaning that they come with scanning artefacts, discolourations, stains, folds, be angled, and have other imperfections. Music scores that were handwritten, of very bad quality, or using non-standard notation were considered out of scope for this work. The selected samples had to be annotated by hand using ScorePad’s current OMR pipeline (Stadelmann et al., 2018). The resulting test set consists of 12 music sheets with a total of 12553 annotations that we name *RealScores*. The annotations are stored in the same format that *DeepScoresV2* (Tuggener et al., 2021) introduced. Due to its limited size, only 61 of the original 136 classes are present. Excerpts from two samples with their corresponding annotations are visible in Figure 2.

In a second step, we sourced a number of “blank” pages from the aforementioned Petrucci Music Library. This was possible because many uploaded music scores would be sourced from completely scanned books, including the front and back covers. Such scans sometimes contain blank pages without any written music, but all the perturbations that naturally occur on sheets of paper. This is a valuable source of real-world noise that can be overlaid with synthetic data. We manually looked through the sourced data

¹Petrucci Music Library (*IMSLP*): https://imslp.org/wiki/Main_Page



Figure 3: Example blank pages.

for suitable blanks, then converted them into pictures and normalised their size to fit with the synthetic data of *DeepScoresV2*. A total of 51 such blank pages were selected – 30 of which have a significant portion of the sheet border visible, and 21 do not. Figure 3 shows six of those pages.

5. Methods

In this section, we present our proposed methods to address domain shift and overconfidence. In Section 5.1 we propose a powerful data augmentation method, in Section 5.2 we present an alternative solution based on UDA, and in Section 5.3 we describe our scheme to produce confidence ratings.

5.1 Input Data Augmentation

We propose a sophisticated data augmentation scheme to address the domain gap that we call *ScoreAug*. With *ScoreAug*, input samples first can be blurred, get salt-and-pepper-like noise, get irregular edges in the border area, be rotated by a small angle, or become augmented with other irregularities not found in a synthetic dataset like *DeepScoresV2*. Additionally, we go one step beyond these algorithmic perturbations

5 METHODS

	Blanks	Scores
Salt and Pepper Noise	-	P_{snp}
No additional Augmentations	-	P_{aug}
Horizontal Flip	50%	-
Vertical Flip	50%	-
Crop and Resize	20%	-
Randomise Brightness	50%	-
Higher Contrast	-	20%
Small Angle Rotation	60%	60%
Additional Brightness	-	40%
Gaussian Blur	-	P_{blur}

Table 2: Probabilities of augmentations as part of *ScoreAug* that can be applied to either the blanks, synthetic scores, or both at the same time. Note that P_{aug} decides how likely any other augmentations (after the salt and pepper noise) will be applied, in order to not only feed *ScoreAugmented* samples to the model. Our final model uses $P_{\text{snp}} = 0\%$, $P_{\text{aug}} = 30\%$, $P_{\text{blur}} = 10\%$.

and complement them by overlaying them on our blank pages from the *RealScores* dataset. Using this combination of augmentation techniques, we aim to bring synthetic data close enough to the real-world domain to train models that generalise to real-world inputs. For a given synthetic input image, we select one out of our set of the 51 blank pages. To increase variability, the blank page and the synthetic data undergo a variety of further augmentations, as shown in Table 2.

To ensure alignment with the transformed image data, the ground-truth bounding boxes also undergo the same transformations. Upon completing these augmentations, the foreground and background are merged by preserving the darker pixel at each position. This means that darker pixels overpower the lighter shades, preserving the dark symbols from the augmented synthetic dataset (the foreground) and replacing the pixels of its white background with the darker pixels from the augmented blank pages (the background). This yields optically similar results to real-world scanned music scores as seen in Figure 4, which can be adapted with hyperparameters (P_{snp} , P_{aug} , P_{blur}) to adjust to one’s needs.

5.2 Unsupervised Adversarial Domain Adaptation

The most common approach to overcome a domain shift is supervised domain adaptation, where densely annotated images are required in the target domain (annotations generally involve instance-level bounding boxes for object detection). Such a solution would require the collection and annotation of a full-scale training dataset consisting of data from the target domain. This approach, therefore, would be tedious and lack the ability to scale, especially for detecting tiny objects in images that are cumbersome to annotate, such as notes in sheet music. Unsupervised



Figure 4: *ScoreAug* examples (top right, bottom row) derived from the same synthetic sample (top left).

domain adaptation (UDA), on the other hand, reduces the expense of annotation by only requiring annotations in the source domain.

Adversarial domain adaptation, which strives to minimise the domain dependency of an object detector via a domain-adversarial loss function utilising a discriminator, is a popular approach for UDA. As highlighted in Tzeng et al. (2017), adversarial domain adaptation is similar to generative adversarial learning, where a generator and discriminator are pitted against each other. For UDA, this concept is used to train a neural network to be unable to differentiate between two domains (in our case synthetic and real-world sheet music images) and ultimately show similar performance on source and target domain samples.

Here, the source domain is the *DeepScoresV2* dataset. For our target domain data, we source non-annotated real-world images from *IMSLP*. Our system consists of a baseline *S²A-Net* (comprising a backbone network $f_{\text{backbone}}^{\theta}$ and an object detector $f_{\text{detect}}^{\theta'}$), a gradient reversal layer and a small domain classifier neural network $f_{\text{domain}}^{\theta''}$. The network weights are denoted by θ, θ' and θ'' respectively. The system is trained using two independent losses, the domain confusion loss L_{domain} and the object detection loss L_{detect} . The following paragraph gives an overview of each component. See Figure 5 for a graphical overview.

The baseline *S²A-Net* is configured as described in

5 METHODS

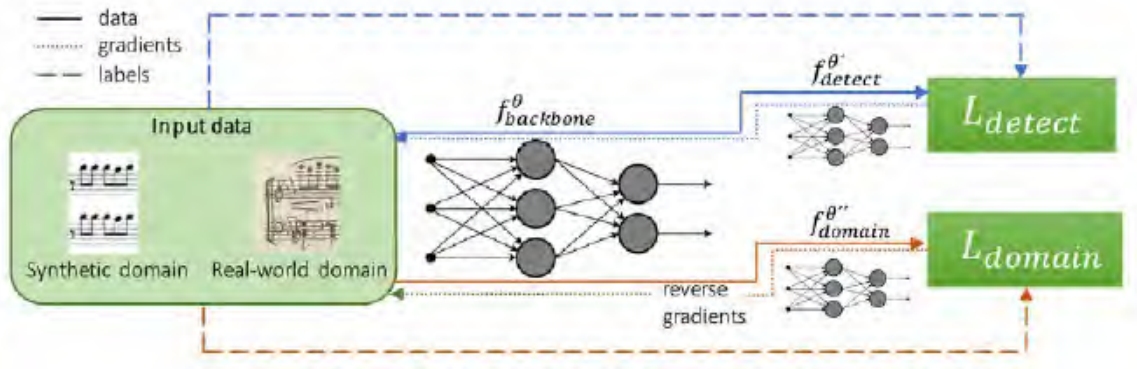


Figure 5: Overview of our UDA system, with data, gradient, and label flow of step (I) shown in orange, of step (II) in green and of step (III) in blue.

Section 3. Here we start with networks that have been fully trained on *DeepScoresV2* to ensure that the network filters are tuned to sheet music. The gradient reversal layer Ganin et al. (2016) can be viewed as a virtual layer in the network that is only active on the backward pass inverting all gradients passing through it. This causes the layers coming after this layer to *maximise* the training loss (in our case $f_{backbone}^{\theta}$ maximizing L_{domain} , causing the backbone embeddings to carry as little information about the domain as possible). $f_{domain}^{\theta''}$ has the job of classifying if an embedding generated by $f_{backbone}^{\theta}$ stems from a data point of the source domain or the target domain. L_{domain} is a binary classification target based on the input domain, as used in GANs Goodfellow et al. (2020). Finally, L_{detect} is the base *S²A-Net* loss to train the whole object detector.

Training the whole system requires the following steps: (I) training $f_{domain}^{\theta''}$ based on L_{domain} (θ'' is getting updated, θ is frozen); (II) use the gradients generated by L_{domain} and propagate them through $f_{domain}^{\theta''}$, applying the gradient reversal layer and propagating the resulting gradients through $f_{backbone}^{\theta}$ tuning θ to *maximise* L_{domain} ; and (III) use labelled samples from the source domain and do a regular *S²A-Net* training step (training $f_{backbone}^{\theta}$ and $f_{detect}^{\theta'}$ based on L_{detect}). Steps (I) and (II) are pitted against each other in an adversarial game, with the goal of "deleting" information that allows the discriminator to differentiate between domains based on the output of the backbone, making the system "domain blind". Step (III) is necessary since the backbone changes and the object detection head needs to adapt to the embeddings accordingly.

Preliminary experiments show that for models without pretraining on the *DeepScoresV2* dataset the resulting UDA do not perform at all. We conjecture that as we are dealing with unlabelled target domain data, it is crucial to learn good representations initially, otherwise the embeddings produced by the backbone

would be too noisy and the domain classifier is unable to learn anything.

While implementing adversarial discriminative domain adaptation, we methodologically distinguish our work from Tzeng et al. (2017) in the following ways:

- We do not use separate networks for source and target domains for efficient sharing of weights.
- We do not fix the weights of our object detection module to allow the object detection module to adapt to the changes in the backbone.
- We do not adopt the asymmetric objective mappings of the feature extractor (in our case the output from *S²A-Net*).

5.3 Confidence Ratings

A music sheet often contains hundreds of musical symbols. Even if OMR software works very reliably, the probability of some misclassifications is high due to the high number of symbols. To identify such misclassifications, it is helpful to analyse the prediction confidence of the model. However, deep learning networks for classification are over-confident because their Softmax layer, which assigns decimal probabilities to each class, tends to push the probabilities either close to 0 or close to 1. Therefore, the model outputs cannot directly be used as a useful measure of confidence Dürr et al. (2020).

We mitigate this issue by using SnapshotEnsemble (Huang et al., 2017) (i.e. multiple predictions) to quantify the predictive uncertainty of our model. This method generates several snapshots (i.e. ensemble members) during training. During inference, each snapshot creates independent predictions of bounding boxes. We use the Weighted Box Fusion (WBF) (Solovyev et al., 2021) algorithm to fuse the bounding boxes. This method constructs the averaged bounding boxes with a corresponding confidence score by utilising the position and confidence scores of all proposed boxes. This overall score can be used as a measurement of the predictive uncertainty.

6 EXPERIMENTS AND RESULTS

<i>DeepScoresV2 dataset</i>	
Model	AP (overlap = 0.25)
Baseline	87.6%
<i>ScoreAug</i>	86.0%
<i>ScoreAug</i> + <i>Finalise</i>	83.3%

<i>RealScores dataset</i>	
Model	AP (overlap = 0.25)
Baseline	36.0%
<i>ScoreAug</i>	56.5%
<i>ScoreAug</i> + <i>Finalise</i>	73.7%

Table 3: The AP for the baseline model and models with *ScoreAug* and *Finalise* data augmentation on the *DeepScoresV2* and the *RealScores* datasets.

6. Experiments and Results

6.1 Input Data Augmentation

Experimental Setup To measure the impact of *ScoreAug*, we trained one baseline model with *ScoreAug* and another without it – each for 2000 epochs. Both models were trained on half-resolution, cropped samples to allow for larger batch sizes and faster convergence. During training, we used a learning rate of $\alpha = 2.5 \cdot 10^{-3}$ throughout and used linear warmup with a ratio of $\frac{1}{3}$ for the first 500 epochs. We observed that the models lack global awareness (e.g. predicting noteheads at the corner of the page), therefore we trained some models an additional 200 epochs on full pages, we denote this step as *Finalise*. During evaluation, we make sure to only consider the results of classes that have at least one positive prediction per model. We evaluate our models using average precision (AP) at an overlap of 25%. We use this unusually low overlap threshold due to the very small object sizes common in MOR, which cause detections that are very usable in practice to often be below the 50% mark.

Results Thanks to *ScoreAug* and *Finalise* we observe an absolute increase in AP of roughly 40% compared to models trained for the same number of epochs and without using both (see Table 3). We observe that on the source Dataset *DeepScoresV2* the performance slightly degrades from 87.6% to 83.3%. However, this is to be expected since we move from a model specifically trained on and for synthetic data to one that can handle a much wider variety of data.

6.2 Unsupervised Adversarial Domain Adaptdation

Experimental Setup Pretraining the *S²A-Net* for UDA showcased impressive results, allowing us to train for relatively few epochs. In our experiments, the pretrained checkpoint had been trained for 250 epochs on the *DeepScoresV2* dataset. We train our UDA pipeline for 30 further epochs. For the domain discriminator, the source domain label is set to 1 and the target domain label is set to 0. The input feature

<i>DeepScoresV2 dataset</i>	
Model	AP (overlap = 0.25)
Baseline	87.6%
UDA	72.4%

<i>RealScores dataset</i>	
Model	AP (overlap = 0.25)
Baseline	36.0%
UDA	48.9%

Table 4: The AP for the baseline model and a model with UDA on *DeepScoresV2* and the *RealScores* dataset.

size is 128, based on the output from *S²A-Net*, and the hidden feature size is 256. Batch normalisation Ioffe and Szegedy (2015) is applied to calculate the mean and standard deviation per dimension over the mini-batches. We use an Adam optimiser Kingma and Ba (2021) with an initial learning rate of 0.01 and constant epoch-driven decay for both targets. We train with a batch size of 4 for both source and target data loaders, which is the maximum batch size our GPUs allowed while keeping the number of samples balanced between domains. In our experiments, for a fair comparison, we follow the same configuration as the baseline model, in terms of *S²A-Net* initialisation and *DeepScoresV2* data loader structures. We limit data augmentations on the *RealScores* data to geometric transformations such as scaling by a factor of 0.5 and random cropping of 1000 by 1000 pixels, both of which are similar to the *DeepScoresV2* data loader samples.

Results Table 4 shows the average precisions for UDA. For the target domain *RealScores* we observe a gain of 12.9% from 36.0% to 48.9%. UDA results in the largest source domain performance loss from 87.6% down to 72.4%. This gain is not quite as impressive as for *ScoreAug*, but we believe it shows the merits of this fully unsupervised approach for MOR. Additionally, the UDA models have been trained only on low-resolution samples to overcome current GPU constraints. It is likely that results would improve in the future with higher resolution images which have generally aided object detection models dealing with tiny objects, such as in MOR.

6.3 Produce Confidence Ratings

Experimental Setup We train different ensemble versions as well as a model not utilising ensembles on the *DeepScoresV2* dataset. We train each model for 1000 epochs and with *ScoreAug*. For the model not utilising ensembles, we use a constant learning rate of $\alpha = 2.5 \cdot 10^{-3}$. For the SnapshotEnsemble models, we start with the same learning rate and decrease it over 500 epochs to $1 \cdot 10^{-5}$ using one single cosine annealing cycle. This rather long cycle is a pretraining of the model before the actual ensemble members are generated. To obtain the ensembles, we train the

6 EXPERIMENTS AND RESULTS

DeepScoresV2 dataset	
Model	AP (overlap = 0.25)
<i>ScoreAug</i>	82.1%
<i>ScoreAug</i> ensemble (10 cycles)	85.6%
<i>ScoreAug</i> ensemble (20 cycles)	87.3%
<i>ScoreAug</i> ensemble (30 cycles)	83.4%

RealScores dataset	
Model	AP (overlap = 0.25)
<i>ScoreAug</i>	37.9%
<i>ScoreAug</i> ensemble (10 cycles)	44.6%
<i>ScoreAug</i> ensemble (20 cycles)	46.7%
<i>ScoreAug</i> ensemble (30 cycles)	47.0%

Table 5: The AP for the model not utilizing ensembles and ensemble models with different cosine annealing cycle lengths on the *DeepScoresV2* and the *RealScores* dataset.

model for 500 additional epochs with shorter cosine annealing cycles over 10, 20, and 30 epochs with learning rates in the range of $1 \cdot 10^{-5} \leq \alpha \leq 7.5 \cdot 10^{-3}$.

After training, the AP for a given overlap of 0.25 is calculated on the test set. In addition, the overlap between snapshots is calculated by using the output from one model as ground truth and the output from a second model as the prediction. We build a set of 10 ensemble members iteratively. We start with an empty set of snapshots. First, the snapshot with the highest AP is added. Afterwards, we add the snapshot which has (i) an overall AP which is max. 5% worse than the AP of our best model; and (ii) has the smallest average overlap with the models which are already added to our set of ensemble members. We repeat this procedure until our set of ensemble members contains 10 snapshots.

The final predictions are the fused boxes generated by the WBF algorithm. The fusion threshold of WBF was set to 0.3 meaning that boxes with the same label and an intersection over union (IoU) of ≥ 0.3 are fused into one box. Since WBF can be used to quantify predictive uncertainty, we use this score to remove predictions with a confidence score below 10% on the *RealScores* dataset. We have found that this improves the prediction quality and reduces false positive rates in particular. In contrast, the predictions on the *DeepScoresV2* dataset are of high quality and no bounding boxes have to be removed based on their confidence score.

Results We have found that ensembles yield better results than a single model. Table 5 shows the AP of the ensemble approaches as well as the AP of the model not utilising ensembles. It can be observed that ensembles improve the AP by up to 5.2 p.p. on the *DeepScoresV2* dataset and up to 9.1 p.p. on the *RealScores* dataset compared to the model without

ensembles. Of the three Cosine Annealing cycle lengths validated, the ensemble with a cycle length of 20 worked best on the *DeepScoresV2* dataset while the ensemble with a cycle length of 30 achieves the highest AP on the *RealScores* dataset. Compared to the results reported in Table 3, the ensemble approaches achieved a lower AP. Since the model's prediction accuracy increases continuously with more training epochs, we suspect that this is due to the fact that the ensembles are trained for only 1000 epochs, while the models in Table 3 are trained for 2000 epochs. However, it is likely that the ensemble would achieve similar or slightly better results, since ensembles typically improve results Dietterich (2000).

Having a high precision and thus a low false-positive rate is particularly important for OMR since it is easier for human annotators to find and label missing annotations than to identify wrong predictions. The confidence ratings can be used to reduce the number of false positive predictions and to increase the precision. We have found that removing predictions with a confidence score below 10% increases precision from 87.8% to 97.2% on the *DeepScoresV2* dataset, and from 35.7% to 41.9% on the *RealScores* dataset respectively. Thus, retaining only predictions with a confidence score bigger than a predefined threshold allows to increase precision at the expense of recall.

Additionally, we assess the confidence ratings visually. Figure 6 shows result excerpts from model outputs with the predictions coloured according to their confidence score. These visualisations can provide useful insights for creating annotations. In accordance with the previous findings, we have observed that analysing the predictions with low confidence is particularly helpful as wrong predictions usually have low confidence.

As in Section 6.1 (c.f. Table 3), we examine the effect of using *ScoreAug* in combination with *Finalise* (i.e. train on full pages) for the ensemble approach. We perform *Finalise* for 50 epochs on each ensemble member obtained. The results with and without *Finalise* are shown in Table 6. The effectiveness of using *Finalise* can be observed particularly clearly on the *RealScores* dataset. When training the ensemble approach for 1000 epochs with *ScoreAug* but without *Finalise*, we achieve an AP of 46.7%. If *ScoreAug* is combined with 50 additional *Finalise* epochs per ensemble member, the AP further improves to 63.6%. *Finalise* thus improves the results not only for single models but also for ensembles. On the source dataset, this once again leads to a small loss in performance from 87.3% to 81.5%. However, *Finalise* in combination with SnapshotEnsembles has the disadvantage that after creating the ensemble members, each member must be fine-tuned separately. This increases the duration of the fine-tuning linearly with the number

7 CONCLUSIONS AND FUTURE WORK



Figure 6: Four cropped visualisation samples of predictions made by an ensemble. The colour of the bounding box indicates the model's confidence (green means high confidence, and red means low confidence). For symbols with a confidence score below 30%, we plot not only the coloured bounding box but also the assigned label as well as the confidence score.

of ensemble members.

A combination of UDA with *ScoreAug* and snapshot ensembles is currently not indicated by the individual results: performance on *RealScores* exceeds 73% using *ScoreAug* + *Finalise* (see Table 3) and reaches beyond 63% when adding confidence ratings via ensembling (see Table 6), but only achieves ca. 49% using UDA (over a baseline of 36%, see Table 4). We do not expect a strong performance boost from a combination, especially since integration is technically uncertain due to the fact that UDA relies on fully pretrained networks and the fragile interplay between steps (I) to (III) that require specific learning rates (see Section 5.2).

7. Conclusions and Future Work

We presented multiple successful avenues towards improving the practical usability of OMR systems. Together, they improve the speed of professional-grade music digitalization on medium-quality scores by more than a factor of 3 over a strong baseline Tuggener et al. (2018b) for high-quality scores. Specifically, 11 minutes per page using the baseline could be reduced to 3.5 minutes on average within the digitalization pipeline of ScorePad AG, which consists of our MOR

<i>DeepScoresV2</i> dataset	
Ensemble (cycle length = 20)	AP (overlap = 0.25)
<i>ScoreAug</i>	87.3%
<i>ScoreAug</i> & <i>Finalise</i>	81.5%

<i>RealScores</i> dataset	
Ensemble (cycle length = 20)	AP (overlap = 0.25)
<i>ScoreAug</i>	46.7%
<i>ScoreAug</i> & <i>Finalise</i>	63.6%

Table 6: The AP for the ensemble trained with a cosine annealing cycle length of 20. The model is trained once with *ScoreAug* only and once with *ScoreAug* in combination with 50 subsequent *Finalise* cycles.

solution mated to a proprietary backend that combines all the information and features a human in the loop correction step. A fully manual transcription by professional musicians would take ca. 40 minutes.

To bridge the domain gap between synthetic datasets and real-world data, algorithmic input augmentation paired with noise sourced from aged real-world documents proved especially fruitful, increasing average detection precision by nearly 50% on the *RealScores* data. In conjunction with *Finalise*, the model performed twice as well as the model trained on synthetic data only.

Unsupervised adversarial domain adaptation showed some promise, outperforming the baseline by 36%. We believe this could be further improved by a UDA method working at very high resolution, to prevent the destruction of fine-grained information in the small patterns of music notation. Both domain adaptation methods had a marginally adverse effect on the performance of the model on synthetic data. In a practical setting, this can be alleviated by employing a data quality classifier and using multiple expert models for high and low-quality data

Using ensembles in combination with weighted box fusion has improved the AP by up to 9.1pp. Besides the better results, ensembles allow us to calculate reliable confidence ratings. These confidence ratings can be used to identify misclassifications, and thus to simplify the manual post-processing of the predictions.

The current models cannot deal with hand-written music scores which could be addressed in the future. Another drawback is the heavy reliance on exact interline scaling, we observed a steep performance drop-off when the interline space is outside of the 8 to 12 pixel range. SnapshotEnsembles creates ensembles without additional training costs by storing snapshots during a single training cycle. The resulting snapshots are fine-tuned separately by using *Finalise* to achieve better performance on real-world data. Training would be more efficient if *Finalise* could be incorporated into the ensemble-generating training cycle and does not have to be done for each ensemble member separately.

REFERENCES

References

- Baró, A., Riba, P., and Fornés, A. (2018). A starting point for handwritten music recognition. In *1st International Workshop on Reading Music Systems*.
- Bellini, P., Bruno, I., and Nesi, P. (2001). Optical music sheet segmentation. In *Proceedings First International Conference on WEB Delivering of Music. WEDELMUSIC 2001*, pages 183–190.
- Calvo-Zaragoza, J., Jr, J. H., and Pacha, A. (2020). Understanding optical music recognition. *ACM Computing Surveys (CSUR)*, 53(4):1–35.
- Calvo-Zaragoza, J. and Rizo, D. (2018a). Camera-primus: Neural end-to-end optical music recognition on realistic monophonic scores. In *19th International Society for Music Information Retrieval Conference, (ISMIR)*, pages 248–255.
- Calvo-Zaragoza, J. and Rizo, D. (2018b). End-to-end neural optical music recognition of monophonic scores. *Applied Sciences*, 8(4):606.
- Castellanos, F. J., Gallego, A.-J., and Calvo-Zaragoza, J. (2021). Unsupervised domain adaptation for document analysis of music score images. In *22nd International Society for Music Information Retrieval Conference, (ISMIR)*, pages 81–87.
- Chen, Y., Li, W., Sakaridis, C., Dai, D., and Van Gool, L. (2018). Domain adaptive faster r-cnn for object detection in the wild. In *IEEE conference on computer vision and pattern recognition (CVPR)*, pages 3339–3348.
- Chiu, C.-C., Sainath, T. N., Wu, Y., Prabhavalkar, R., Nguyen, P., Chen, Z., Kannan, A., Weiss, R. J., Rao, K., Gonina, K., Jaitly, N., Li, B., Chorowski, J., and Bacchiani, M. (2018). State-of-the-art speech recognition with sequence-to-sequence models. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4774–4778.
- Chowdhury, A. and Vig, L. (2018). An efficient end-to-end neural model for handwritten text recognition. In *British Machine Vision Conference 2018, (BMVC)*, page 202.
- Ciregan, D., Meier, U., and Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. In *2012 IEEE conference on computer vision and pattern recognition (CVPR)*, pages 3642–3649. IEEE.
- Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., and Le, Q. V. (2019). Autoaugment: Learning augmentation strategies from data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 113–123.
- Dalitz, C., Droettboom, M., Pranzas, B., and Fujinaga, I. (2008). A comparative study of staff removal algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(5):753–766.
- Dietterich, T. G. (2000). Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15.
- Durasov, N., Bagautdinov, T., Baque, P., and Fua, P. (2021). Masksembles for uncertainty estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13539–13548.
- Dürr, O., Sick, B., and Murina, E. (2020). *Probabilistic Deep Learning: With Python, Keras and TensorFlow Probability*. Manning Publications.
- Elezi, I., Tuggenier, L., Pelillo, M., and Stadelmann, T. (2018). Deepscores and deep watershed detection: current state and open issues. In *1st International Workshop on Reading Music Systems*.
- Fujinaga, I. (2004). Staff detection and removal. In *Visual Perception of Music Notation: On-Line and Off Line Recognition*, pages 1–39.
- Gallego, A.-J. and Calvo-Zaragoza, J. (2017). Staff-line removal with selectional auto-encoders. *Expert Systems with Applications*, 89:138–148.
- Ganin, Y. and Lempitsky, V. (2015). Unsupervised domain adaptation by backpropagation. In *international conference on machine learning (ICML)*, pages 1180–1189.
- Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. (2016). Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, 63(11):139–144.
- Gustafsson, F. K., Danelljan, M., and Schon, T. B. (2020). Evaluating scalable bayesian deep learning methods for robust computer vision. In *IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 318–319.
- Hajič, J. and Pecina, P. (2017). The muscima++ dataset for handwritten optical music recognition. In *14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, pages 39–46.
- Hajic Jr, J., Dorfer, M., Widmer, G., and Pecina, P. (2018). Towards full-pipeline handwritten omr with musical symbol detection by u-nets. In *19th International Society for Music Information Retrieval Conference, (ISMIR)*, pages 225–232.
- Han, J., Ding, J., Li, J., and Xia, G.-S. (2021). Align deep features for oriented object detection. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–11.
- He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *IEEE conference on computer vision and pattern recognition (CVPR)*, pages 770–778.
- Huang, G., Li, Y., Pleiss, G., Liu, Z., Hopcroft, J. E., and Weinberger, K. Q. (2017). Snapshot ensembles:

REFERENCES

- Train 1, get M for free. In *5th International Conference on Learning Representations, (ICLR)*.
- Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning (ICML)*, pages 448–456.
- Kingma, D. P. and Ba, J. (2021). Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR*.
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90.
- Lehner, A., Gasperini, S., Marcos-Ramiro, A., Schmidt, M., Mahani, M.-A. N., Navab, N., Busam, B., and Tombari, F. (2022). 3d-vfield: Adversarial augmentation of point clouds for domain generalization in 3d object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17295–17304.
- Li, Y.-J., Dai, X., Ma, C.-Y., Liu, Y.-C., Chen, K., Wu, B., He, Z., Kitani, K., and Vajda, P. (2022). Cross-domain adaptive teacher for object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7581–7590.
- Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). Feature pyramid networks for object detection. In *IEEE conference on computer vision and pattern recognition (CVPR)*, pages 2117–2125.
- Mateiu, T. N., Gallego, A.-J., and Calvo-Zaragoza, J. (2019). Domain adaptation for handwritten symbol recognition: A case of study in old music manuscripts. In *Iberian Conference on Pattern Recognition and Image Analysis*, pages 135–146.
- Nguyen, A., Yosinski, J., and Clune, J. (2015). Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *IEEE conference on computer vision and pattern recognition (CVPR)*, pages 427–436.
- Pacha, A. and Calvo-Zaragoza, J. (2018). Optical music recognition in mensural notation with region-based convolutional neural networks. In *19th International Society for Music Information Retrieval Conference, (ISMIR)*, pages 240–247.
- Pacha, A., Choi, K.-Y., Coüasnon, B., Riquebourg, Y., Zanibbi, R., and Eidenberger, H. (2018a). Handwritten music object detection: Open issues and baseline results. In *2018 13th IAPR International Workshop on Document Analysis Systems (DAS)*, pages 163–168. IEEE.
- Pacha, A., Hajič, J., and Calvo-Zaragoza, J. (2018b). A baseline for general music object detection with deep learning. *Applied Sciences*, 8(9):1488.
- Pugin, L. (2006). Optical music recognition of early typographic prints using hidden markov models. In *Proceedings of the 7th International Society for Music Information Retrieval Conference (ISMIR)*, pages 53–56.
- Rebelo, A., Fujinaga, I., Paszkiewicz, F., Marcal, A. R., Guedes, C., and Cardoso, J. S. (2012). Optical music recognition: state-of-the-art and open issues. *International Journal of Multimedia Information Retrieval*, 1:173–190.
- Sato, I., Nishimura, H., and Yokoi, K. (2015). Apac: Augmented pattern classification with neural networks. *arXiv preprint arXiv:1505.03229*.
- Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61:85–117.
- Simmler, N., Sager, P., Andermatt, P., Chavarriaga, R., Schilling, F.-P., Rosenthal, M., and Stadelmann, T. (2021). A survey of un-, weakly-, and semi-supervised learning methods for noisy, missing and partial labels in industrial vision applications. In *2021 8th Swiss Conference on Data Science (SDS)*, pages 26–31.
- Smith, L. N. (2017). Cyclical learning rates for training neural networks. In *2017 IEEE winter conference on applications of computer vision (WACV)*, pages 464–472.
- Solovyev, R., Wang, W., and Gabruseva, T. (2021). Weighted boxes fusion: Ensembling boxes from different object detection models. *Image and Vision Computing*, pages 1–6.
- Stadelmann, T., Amirian, M., Arabaci, I., Arnold, M., Duivesteijn, G. F., Elezi, I., Geiger, M., Lörwald, S., Meier, B. B., Rombach, K., et al. (2018). Deep learning in the wild. In *IAPR Workshop on Artificial Neural Networks in Pattern Recognition*, pages 17–38.
- Toyama, F., Shoji, K., and Miyamichi, J. (2006). Symbol recognition of printed piano scores with touching symbols. In *18th International Conference on Pattern Recognition (ICPR)*, volume 2, pages 480–483.
- Tuggener, L., Elezi, I., Schmidhuber, J., Pelillo, M., and Stadelmann, T. (2018a). Deepscores-a dataset for segmentation, detection and classification of tiny objects. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3704–3709.
- Tuggener, L., Elezi, I., Schmidhuber, J., and Stadelmann, T. (2018b). Deep watershed detector for music object recognition. In *19th International Society for Music Information Retrieval Conference (ISMIR)*, pages 271–278.
- Tuggener, L., Satyawat, Y. P., Pacha, A., Schmidhuber, J., and Stadelmann, T. (2021). The deepscoresv2 dataset and benchmark for music object detection. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 9188–9195.
- Tzeng, E., Hoffman, J., Saenko, K., and Darrell, T. (2017). Adversarial discriminative domain

REFERENCES

- adaptation. In *IEEE conference on computer vision and pattern recognition (CVPR)*, pages 7167–7176.
- van der Wel, E. and Ullrich, K. (2017). Optical music recognition with convolutional sequence-to-sequence models. In *18th International Society for Music Information Retrieval Conference, (ISMIR)*, pages 731–737.
- von Oswald, J., Kobayashi, S., Sacramento, J., Meulemans, A., Henning, C., and Grewe, B. F. (2021). Neural networks with late-phase weights. In *9th International Conference on Learning Representations, (ICLR)*.
- Wen, Y., Tran, D., and Ba, J. (2020). Batchensemble: an alternative approach to efficient ensemble and lifelong learning. In *8th International Conference on Learning Representations, (ICLR)*.
- Wenzel, F., Snoek, J., Tran, D., and Jenatton, R. (2020). Hyperparameter ensembles for robustness and uncertainty quantification. In *34th International Conference on Neural Information Processing Systems (NeurIPS)*, pages 6514–6527.
- Xia, Y., Zhang, J., Jiang, T., Gong, Z., Yao, W., and Feng, L. (2021). Hatchensemble: an efficient and practical uncertainty quantification method for deep neural networks. *Complex & Intelligent Systems*, 7:2855–2869.
- Zhu, X., Liu, Y., Qin, Z., and Li, J. (2017). Data augmentation in emotion classification using generative adversarial networks. *arXiv preprint arXiv:1711.00648*.
- Zhu, X., Pang, J., Yang, C., Shi, J., and Lin, D. (2019). Adapting object detectors via selective cross-domain alignment. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 687–696.

Originally published in Proceedings of the 11th IEEE Swiss Conference on Data Science (SDS'24), Zurich, Switzerland, May 31, 2024, DOI 10.1109/SDS60720.2024.00027.

Automated process monitoring in injection molding via representation learning and setpoint regression

✉ Peng Yan^{*¶}, ✉ Ahmed Abdulkadir^{*¶}, ✉ Giulia Aguzzi[†], ✉ Gerrit A. Schatte[†],
✉ Benjamin F. Grewe[‡], and ✉ Thilo Stadelmann^{*§}

{yanp, abdk, stdm}@zhaw.ch, {giulia.aguzzi,gerrit.schatte}@kistler.com, bgrewe@ethz.ch

^{*}Centre for Artificial Intelligence, ZHAW School of Engineering, Winterthur, Switzerland

[†]Innovation Lab, Kistler Instrumente AG, Winterthur, Switzerland

[‡]ETH AI Center, ETH Zurich, Zurich, Switzerland

[§]Fellow, ECLT (European Centre for Living Technology, Venice, Italy) and Senior Member, IEEE

[¶]These authors contributed equally to this work and share the first authorship

Abstract—Online process monitoring is essential to detect failures and respond promptly in automated industrial processes such as injection molding. Traditional systems rely on experienced operators manually defining operational boundaries around a reference signal. We propose a data-driven representation that auto-tunes the sensitivity to a pre-set specificity threshold and automatically detects anomalies alongside interpretable indices that help identify root causes. Our automated system achieved an average AUC of 0.998 and detected 100 percent of the anomalies with the proposed dynamic calibration of the data-driven embedding method. The dynamic calibration, which accounted for drift, boosts the average specificity from 0.362 to 0.869. The outputs also indicate the direction and relative magnitude of characteristic deviations caused by machine parameters, including holding pressure, mold temperature, and injection speed. The AI-derived process boundaries are superior to manual annotation in tested real-world production environments.

Index Terms—anomaly detection, time series, variational auto-encoder, root-cause analysis, explainable AI, transfer learning

I. INTRODUCTION

Injection molding is a cornerstone in mass-producing plastic parts with high precision [1]. This process involves injecting molten plastic into molds, forming the final products. Monitoring time series from in-mold sensors can help detect abnormal behaviors in real time [2]. For example, the transient temperature curve in each mold's cavity measured via an infrared sensor or thermocouple is expected not to change in stable operations [3]. Thus, the time series of sensors acquired under normal operation represents a reference for detecting deviations or anomalies. Early detection of anomalies or deviations can trigger timely intervention that lowers production costs [4]. Traditionally, anomaly detection has relied on the expertise of operators when analyzing complex data like pressure curves. However, this manual approach has limitations: it's labor-intensive and inconsistent due to varying operator expertise. It needs to be improved. Compounding the need for improvement is the declining number of apprenticeships for shopfloor workers in the European and US manufacturing ecosystems [5]. This is expected to result in fewer personnel operating more machines on the shop floor in the following

This work is supported by Innosuisse grant 62174.1 IP-ENG "DISTRAL".

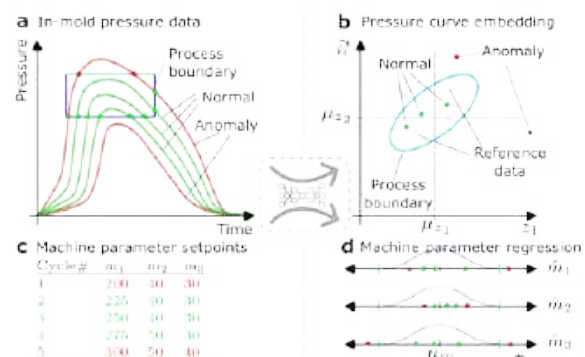


Fig. 1: Overview of this work: Manual process boundaries in the traditional approach (a) that depend on machine settings (c) and other factors and proposed data-driven process boundaries (b and d). (a) Examples of pressure curves are traditionally classified as *normal* (green) or *abnormal* (red) using evaluation labels (the rectangle with three blue and one cyan edge on the top). A normal curve must pass through at least one blue line and not go through a cyan line. A pressure curve is defined as an *anomaly* if it fails to meet any specified requirement. (b) Our deep learning-based approach maps the pressure curves into a two-dimensional embedding for automated analysis. Given a set of reference embedded curves (purple crosses), the adaptive process boundaries (cyan ellipse in b) are computed automatically. (c) Machine parameter setpoints control the injection molding process, and thus impact the part quality. (d) The neural network is also used to estimate machine parameters and the direction and magnitude of the standardized calibrated data distribution assist in detecting root causes.

decades. Increasing automation in the monitoring process aims to improve efficiency, accuracy, and product quality [6], [7].

In the injection molding process, sensors are installed in the mold to capture process signals during production. Among those signals, the pressure time series is a fingerprint of the injection molding process as it determines the crystallization

of the molten substrate in the mold [8]. Its monitoring contains information about the process stability and, implicitly, the part quality [9] (or its deviation from the desired state) without measuring the dimensions, weight, and appearance of every part. It is, therefore, suitable for processes that need monitoring but not a 100%-part inspection. In industrial solutions like ComoNeo [10], operators traditionally set so-called evaluation objects (EOs) on these curves manually (see Fig. 1a) to define process boundaries. The EOs are set before the production nominally starts for each specific process and part and depend on the operator's experience with different materials, shapes, cycle times, or typical process parameters. They are hence highly subjective and their setup and any intervention, once an anomaly is detected, relies on the individual highly skilled and experienced professional. The location and extent of the boundaries determine the sensitivity and specificity of the anomaly detection process concerning certain types of anomalies. For example, a malfunction of the heating system of the substrate will impact the pressure curves differently than a change in the substrate. In theory, setting multiple EOs allows for fine-tuning the detection precisely, but in practice, possible failure scenarios with their impact are unknown. Thus, deeper automation is required to remove the manual definition of process boundaries and help identify root causes.

Machine learning, deep learning (DL), and statistical modeling offer transformative potential in this field [11]. By adopting a data-centric approach, we can surpass limitations imposed by manual tools [12], [13], allowing operators to focus on timely interventions and quality control. DL algorithms, trained on extensive injection molding data, can effectively represent pressure curves. Especially, variational autoencoder (VAE) [14] is a powerful DL framework for learning low-dimensional interpretable latent representations in an unsupervised manner [15] due to the unavailability of sufficient labeled data in the industrial setting [16]. Applying such DL models and using statistical methods for anomaly detection in new processes promises tuneable sensitivity, high accuracy in detecting anomalies, and a more efficient, consistent, and predictive monitoring strategy [17] while triggering appropriate interventions to maintain product quality and process efficiency [18]. Furthermore, utilizing a DL model that incorporates domain-specific knowledge can be instrumental in identifying these anomalies and infer the underlying causes.

In this paper, our main contribution is a practical solution for automated anomaly detection and machine parameter prediction for injection molding processes without feature engineering: (1) a novel VAE-based model that represents transient 1D sensor signals in a 2D latent space with robust out-of-distribution embedding capabilities which is used to detect deviations (anomalies) from reference curves representative of a stable process, and utilizes additional neurons that capture changes in machine parameter setpoints for interpretable outputs; (2) a dynamic calibration process that automatically determines the operational bounds of the process for anomaly detection with the ability to account for drifts; (3) prediction of set machine parameters to assist the root-cause identification.

We assess our approach's effectiveness with real-world data from injection molding production in which data originated from different production environments, achieving a remarkable AUC score of 0.998.

II. RELATED WORK

DL-based methods have been widely used for the analysis of industrial time series data as recently surveyed by Yan et al. [19]. These methods can be divided into reconstruction-based, forecasting-based, and other statistical methods. In deep reconstruction-based methods, the anomalies are recognized by comparing the deviation of the reconstructed time series sequence and the actual sequence using a defined threshold [20]. In contrast, for forecasting-based anomaly detection, only the forecasted sequence is compared with the ground truth sequence [21]. Most of the related anomaly detection methods assume that testing and training data follow the same data distribution, although by design, the "normal" state is unknown *a priori*. Regarding model architecture, Tuggenier et al. have shown that the best-performing model architecture depends much more on the actual data set at hand than on the current state-of-the-art method on public benchmarks [22]. In this paper, we build on simple but well-tuned reconstruction-based models based on preliminary experiments.

Real-world industrial processes are dynamic and can be affected by a variety of production conditions [23]. Taking the injection molding process as an example, ambient environment, machine wear and tear, and other variables can exert an unknown but sometimes high influence on the manufacturing process. These conditions can then lead to a drift of the observed control data during production [24], [25]. The aforementioned anomaly detection methods might not be applicable. Various approaches have been proposed to deal with domain shift issues, such as transfer learning and domain adaptation. For instance, Saurav et al. [26] proposed a temporal model based on recurrent neural networks for time series anomaly detection to account for sudden or regular changes in normal behavior. The model is updated incrementally as new data becomes available and is capable of adapting to the changes in the data distribution. Instead of adapting directly to the new domain, Yang et al. seek to learn a domain invariant representation [27]. They train a DL model to extract a domain-invariant representation from normal data from the source domain and a limited number of normal data in the target domain. The domain-invariant representation is achieved by adversarial learning. Apart from that, transfer learning approaches have been applied in injection molding to transfer knowledge from one or more source setups to solve tasks in new setups [28], [29]. However, such approaches analyze machine parameters instead of sensory data collected directly from the manufacturing process, hence the transfer learning's effectiveness needs to be further validated. In this paper, we develop an auto-calibration mechanism based on statistical methods to mitigate data drift from sensor data while adding machine parameters for root cause analysis.

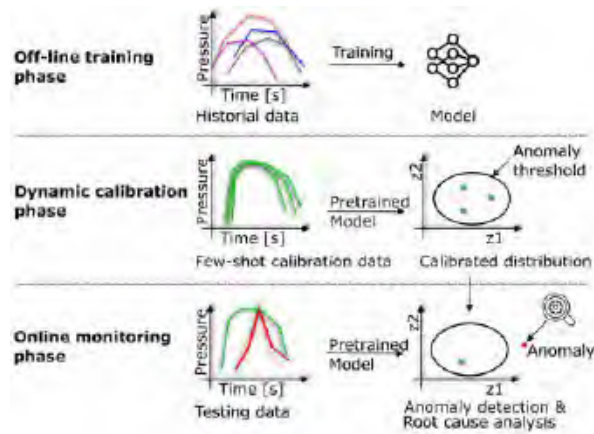


Fig. 2: Proposed data driven anomaly detection. A large data set of pressure curves is used to train a model to map pressure curves to a low-dimensional representation z (Here we only show the 2D representations of the pressure curve). Once the molding process is stable, a small sample of reference curves is mapped into the latent space to calibrate the process boundary. During online monitoring, an anomaly event is triggered if a data point is mapped to a position outside the anomaly threshold (calibrated process boundary).

Explainability and interpretability of DL models enable the quantification of the contribution of input features to the prediction results [30]. Alternatively, handcrafted features from the injection molding process have been investigated in [31]. An important factor analysis and a random forest model is used to identify the most influential features in cavity pressure and temperature data.

However, the relevant features may differ for different production conditions. Determining them requires expert knowledge and manual intervention and could miss other relevant information from the manufacturing processes. In this paper, we use the rich in-mold pressure signals without manual feature engineering, adding machine parameters to assist root cause analysis.

III. PROPOSED METHODS

A. Overall pipeline

Fig. 2 depicts the overall pipeline of the proposed methods for data-driven anomaly detection. It consists of off-production representation learning, in-production dynamic calibration, and online monitoring. In the off-production training phase, historic pressure curves from various batches are used to train a useful low-dimensional representation of pressure curves. This is possible because the characteristic shape of the relevant curves is similar across productions (rapid ascent and a slower descent) but varies in the location of the peak and slopes.

Once the process is stable and produces the parts of the desired quality, the latent representation z_p of a small number

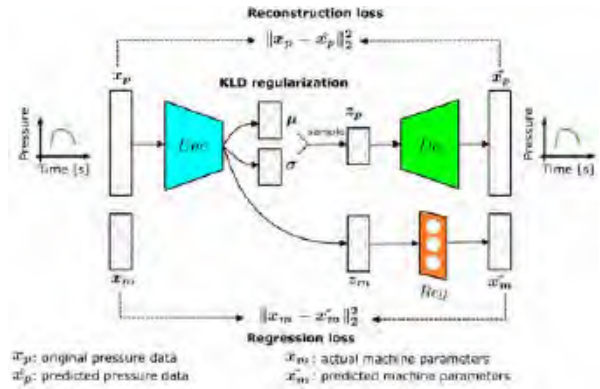


Fig. 3: Model architecture of the autoencoder with machine parameter prediction. The model consists of encoder Enc , decoder Dec , and regressor Reg . Training is driven by reconstruction loss of the pressure curves, regression loss of machine parameters, and distribution regularizer over z_p . See text for further details.

$N_C = \{10 \dots 100\}$ of reference cycles is extracted using the pre-trained model. The embedded low-dimensional reference data are then used to estimate their distribution parameters. Under stable conditions, normal data from subsequent cycles are expected to follow the estimated distribution, while that of the anomalies is expected to deviate from it.

Analogous and in parallel, a predictor for machine parameter setpoints z_m is trained (see Figure 3). Deviations from the calibrated reference distribution of these predictions can be used for root cause identification.

B. Deep learning architecture

We employ an auto-encoder that reconstructs the input through a bottleneck layer [32] to leverage unlabeled data and learn a useful representation. The network is implemented as a VAE with convolutional layers and a two-dimensional variational bottleneck without tied weights in the four hidden layers of the encoder and decoder. Fig. 3 shows our proposed network architecture, which consists of an encoder, a decoder, and a regressor.

Encoder. The encoder receives a pressure time series $p = (p_1, \dots, p_D)$ and maps it into μ, σ, z_m , where μ, σ represent the mean and variance of the estimated 2D Gaussian distribution, and z_m denotes the latent representation for 3 machine parameters. In summary, we can express the mapping as $\mu, \sigma, z_m = Enc(p)$. Moreover, z_p is sampled from the learned Gaussian distribution, given by $z_p = \mu + \varepsilon \cdot \sigma$, where $\varepsilon \sim \mathcal{N}(0, I)$.

Decoder. After encoding the pressure signal into a low-dimensional latent representation, we use the decoder Dec to reconstruct the pressure signal with the original dimension from z_p , which can be denoted as $\hat{p} = Dec(z_p)$.

Machine parameter regression. Three non-variational neurons are added to the VAE and connected to the bottleneck

layer. They predict setpoints of holding pressure, injection speed, and molding temperature via an extra layer $\hat{m}_i = \text{Reg}_i(z_{m_i})$. This extra linear mapping between the intermediate representation z_m and the output $\hat{m}$ allows production-batch-specific scaling and shift factors during training. For new batches (e.g. using different materials), the mapping between the pressure curve and machine parameters is expected to be proportional. For our application of detecting outliers and the direction of deviations in the machine parameter predictions, we use z_m for the analysis.

C. Loss function

The VAE minimizes the mean squared reconstruction error between the predicted and original pressure data. The two-dimensional latent variables are regularized to follow a bivariate standard Gaussian distribution using the KL divergence. Reconstruction loss and regularization are formulated as

$$L_{rec} = \frac{1}{ND} \sum_{n=1}^N \sum_{i=1}^D (p_i^n - \hat{p}_i^n)^2 \quad (1)$$

$$L_{kld} = -\frac{1}{2N} \sum_{n=1}^N \sum_{i=1}^2 [1 + \log(\sigma_{n,i}^2) - \mu_{n,i}^2 - \sigma_{n,i}^2] \quad (2)$$

where N and D are the sample size and the number of time stamps in the pressure time series, and p_i^n and $\hat{p}_i^n$ are the original and predicted pressure signal at time step i for n -th time series sample. In addition, we apply L2 loss between the predicted and original machine parameters:

$$L_{reg} = \frac{1}{NC} \sum_{n=1}^N \sum_{i=1}^C (m_i^n - \hat{m}_i^n)^2 \quad (3)$$

where C is the number of machine parameters, and m_i^n and $\hat{m}_i^n$ are the original and predicted machine parameters for n -th time series sample. The overall objective is to minimize

$$Loss = L_{rec} + \omega_{kld} \cdot L_{kld} + \omega_{reg} \cdot L_{reg} \quad (4)$$

where ω_{kld} and ω_{reg} are weights to balance losses. We also apply L2 regularization for the model weights during training.

D. Calibration and setting of process boundaries

Calibration is necessary to set meaningful process boundaries of an unknown process automatically. Using a few numbers (N_C) of reference curves from a stable process, the system fits a multivariate Gaussian distribution on the bottleneck node and outputs means μ . Additionally, it estimates an univariate Gaussian distribution for each unscaled machine-parameter neuron output z_{m_i} .

Given the empirical mean of the reference sample distribution, the distance of a new sample to the center of the reference distribution mean can be interpreted as an anomaly indicator. Specifically, the larger this distance, the more abnormal the new sample is considered. However, the scale of the Euclidean distance, especially in the multi-dimensional latent space, is meaningless. We overcome this by incorporating the variances and relating the resulting statistical measures analytically to

the cumulative distribution function as follows: The distance standardized by either the covariance matrix in the multivariate distribution or the variance in the univariate distribution yields the Mahalanobis distance [33], respectively. Based on the Mahalanobis distance to the reference distribution d , a chi-squared distribution $\chi^2(d, k)$ is employed to express the fraction of samples i from the distribution where $d_i < d$, with $k = |\mu|$ representing the degree of freedom. An operational bound B can thus be set such that a sample is considered an anomaly if $\chi^2(d, k) > B$, where $(1 - B)$ is the expected rate of false positives given a stationary process.

Since the machine parameters express physical properties that impact the process and are characterized by one-dimensional distributions, the system retains the sign of the relative deviation from the mean of the distribution. This facilitates root cause analysis, suggesting the direction of a parameter change as an intervention to recover a stray process.

When a process is non-stationary, meaning that the mean of the distribution gradually shifts over time, or if the noise exhibits heteroscedastic behavior (variance changes with time), the parameters of the initial calibration eventually lose their validity. This can happen due to external factors such as changes in environmental conditions like room temperature or humidity. We propose a dynamic update of the distribution parameters based on the latest N_C cycles.

IV. EXPERIMENTAL SETUP AND EVALUATION

In this section, we describe the experimental setup and evaluation results, including the dataset, training procedure, and metrics.

A. Data: collection, handling, and preprocessing

IMM datasets. The injection molding machine (IMM) data is pooled from non-systematically acquired and incompletely labeled historical data provided by three vendors of injection molding systems. The resulting datasets were classified based on the availability of anomaly labels and machine parameters (see D0–D4 in Tab. I). Each mold has one to 8 Kistler pressure sensors attached to each cavity. The sensors measure the pressure in the cavity during each cycle and each sensor reports its pressure reading independently. The data are acquired at 16kHz for a cycle period between 0.6 and 20 seconds, depending on the size and shape of the products. After the acquisition, the time series signals are downsampled to a maximum of 1,000 measurement points with irregular time intervals. The time series signals (per cavity/sensor) are stored in a database with various meta-data such as the time of day, mold ID, cavity ID, and a binary quality label. The labels are derived from EO, visual inspection, or physical measurements. The 5 datasets D0–D4 represent common production data but cover only a small fraction of the possible variability in mold, machine, part size, substrate, etc. In addition, a total of 540 pressure readings contained in two datasets (D5–D6) have been acquired within a controlled experimental design where we systematically varied the 3 machine parameters using 3

TABLE I: IMM datasets characteristics used for training, calibration, and testing.

Dataset	Sample size (train/val)	Cavity count	Cycle duration (s)	Anomaly rate	Different machine settings count (train/val)
D0	5,208 / 1,240	8	6	assumed 0%	18 / 5
	Sample size (cal/test)				Different machine settings count (cal/test)
D1	70 / 2,905	7	1.5	5.50%	unknown
D2	80 / 5,104	8	2	0.78%	unknown
D3	80 / 4,240	8	2.5	2.08%	unknown
D4	80 / 3,024	8	1.5	1.59%	unknown
D5	10 / 260	1	10	assumed 0%	1 / 26
D6	10 / 260	1	10	assumed 0%	1 / 26

distinct values for each, yielding 10 cycles for each of the $3^3 = 27$ possible combinations.

Training and validation data. Out of a larger collection, we construct the unlabeled dataset D0 by pre-selecting pressure curves that visually appear similar to remove any environmental influence on the data apart from the known machine parameter settings. This yields 6,448 pressure data with 23 machine settings out of 44 in the overall collection. Then, data from random 18 machine settings is used for training and the remaining 5 for validation.

Calibration and testing data. We have two groups of testing data for evaluating anomaly detection and machine parameter indication, respectively, using D1–D4 to evaluate time series anomaly detection performance and D5–D6 to evaluate the performance of machine parameter indication. For D1–D4, we only keep the data with valid EO labels, i.e., we keep the data generated when a specific number of consecutive normal samples are obtained. In D1, we identified an invalid cavity that recorded noise and excluded the corresponding data in the study. In all experiments, we use a calibration window size (number of consecutive normal pressure curves) of $N_C = 10$. For the dynamic calibration, we re-calculate the distribution parameters for each cycle with the previous $N_C = 10$ cycles.

Data preprocessing. Since the original time series data is sampled with irregular time intervals, we interpolate the pressure time series data into the same length (200) with equidistant time intervals. More specifically, the irregularly sampled time series data are resampled in 200 samples with regular intervals between 0 and 10 seconds. Additionally, we normalize pressure data by dividing by 1,000 and subtracting 0.5. This roughly brings the observed data between -0.5 and 0.5 . The inverse operation is applied to the network output to recover the original location and scale. This ensures a uniform scale across training, validation, calibration, and testing.

B. Training process

We train the model using stochastic gradient descent with batch size 16 for 200 epochs. Gradient updates are performed with the Adam optimization procedure [34]. The coefficients for the losses in Eq. (4) are set as $\omega_{kld} = 100, \omega_{reg} = 10$, balancing the terms such that they are of roughly equal importance. We train the model with early stopping when the validation loss does not decrease for 5 consecutive epochs.

C. Evaluation metrics

To evaluate the anomaly detection performance, we obtain the confusion matrix of the anomaly detection based on the actual (EO-based) and predicted (using process boundaries estimated from calibration data) labels of the testing dataset. Based on the confusion matrix, we then calculate the recall/true positive rate (TPR) and specificity. TPR measures the ratio of correctly classified normal samples to all actual normal samples. Specificity measures the ratio of the predicted negatives to the actual negatives. Furthermore, leveraging the Mahalanobis distance between the testing data and the estimated Gaussian distribution, we compute the area under the receiver operating characteristic curve (AUC score) as the main metric to assess the performance of anomaly detection.

D. Results

Overview. In the experiments, D5 and D6 share similar experimental settings across different products. For brevity, we only show one arbitrary dataset per figure as both datasets yield the same conclusion. Fig. 4 shows a comprehensive illustrative analysis of data from D6. The pressure data are mapped into the latent space to obtain μ and z_m . The green color represents calibration data (assumed normal cases), and the ellipses represent different estimated distributions for different confidence levels. A suitable operational bound is set to classify abnormal samples (we assume data from machine settings other than calibration data are abnormal). We also analyze the machine parameter indication by comparing the latent representation for the machine parameters. Note that the reconstructed signals and the proportionally scaled machine parameters are not of interest and, therefore, not reported here.

Anomaly detection. We provide qualitative and quantitative analyses for anomaly detection based on the deviation from the calibrated reference distribution of z_p . Quantitative results are summarized in Tab. II. AUC scores are reported as average over all cavities of each dataset. In addition, we compute recall, specificity, and odds ratio at the operational bound of $B=0.95$. The odds ratio is computed as the ratio between precision and *a priori* anomaly rate. We compare results between one-time calibration and dynamic calibration. The results show that dynamic calibration outperforms one-time calibration in most instances since it consistently adapts to the data drift over time. As each cavity is calibrated separately and the anomaly value is expressed as the Mahalanobis distance to

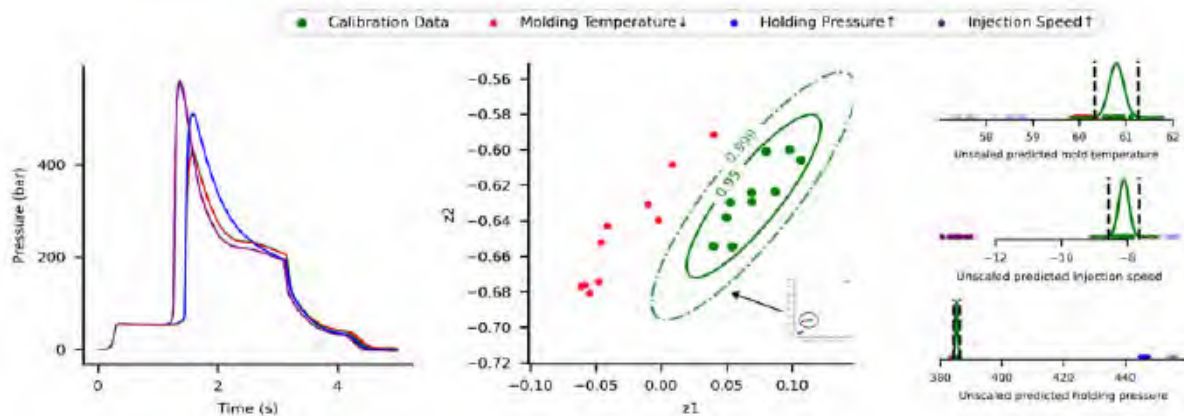


Fig. 4: **Left:** Pressure curves of 4 machine settings from D6, using 10 reference cycles from one machine setting as calibration data (green) and the rest for evaluation, where only a single machine parameter has been changed. **Middle:** Process boundaries at $B=0.95$ (solid line) and $B=0.999$ (dashed line). The small zoomed-out figure shows that the latent representations from increasing injection speed and holding pressure significantly deviate from the calibrated distribution. **Right:** Machine parameter predictions superimposed on the calibrated distribution (dotted lines indicate a deviation of 99.9% from the mean). It is sensitive to changes in the set values (e.g., for the red curves with set lower molding temperature, the molding temperature is indeed predicted lower, pointing at the anomaly's root cause as well as at an intervention). It is however less specific (i.e., a lower molding temperature is also predicted for the blue and purple curves, although temperature was kept constant there). This lack of disentanglement is partly due to the extreme nature of the parameter variations here as well as to actual physical couplings.

the cavity-specific distribution's center, the values are directly comparable across cavities.

Machine parameter prediction. Datasets D5–D6 are included for the evaluation of the prediction of the machine parameters because of their $3 \times 3 \times 3$ design in which each of the 3 machine parameters is changed threefold while keeping the others constant. The prediction of the holding pressure and the injection speed are in a positively linear relation, as shown in Fig. 5. The effect of the mold temperature is smaller than that of holding pressure and injection speed. For example, at 400 bar holding pressure and 30 mm/s injection speed, the prediction of the temperature is unreliable.

V. DISCUSSION AND CONCLUSIONS

Summary. This study marks significant progress in the practical automation of injection molding process monitoring. It introduces a straightforward and effective method for domain adaptation and root cause identification, addressing prior challenges in transfer learning and unrealistic data requirements. We built on the state of the art in representation learning, employed statistical modeling, and injected domain knowledge to overcome three main limitations in the automation of injection molding anomaly detection and root cause analysis with a novel model and process. Specifically, the dependence on experienced personnel and manual configuration of the detection process has been reduced by using a variational auto-encoder-based representation that is dynamically calibrated with reference data during production. An objective choice of acceptable false positive rate replaces

the subjective EOs. The designed machine-parameter-sensitive activations facilitate rule-based root cause identification that increases the practical value of the output. As such, the proposed system meets the needs of a self-calibrating system for anomaly detection. Our approach is implemented in the context of injection molding processes, but it also applies to other transient sensor data in which a deviation from a “reference” indicates an anomaly. The solution automatically sets process boundaries in a low-dimensional and explainable representation of pressure curves in a data-centric way. The detection process can be dynamically re-calibrated if desired. The principle is not limited to 1D time-series signals, but the representation-learning based framework generalizes to multi-dimensional data if the reference data is consistent and enough data is available. The same dynamic calibration process can be applied elsewhere as long as smooth data drift occurs. This conceptual advance has been filed for a European patent [35].

Discussion. We evaluated the feasibility of automatically setting meaningful process boundaries in unseen process data based on a small number of reference cycles. Data were acquired with sensors of a single modality, but the principle applies to other transient signals. The test data were distinct from the training data, most notably in cycle duration, which was 6 seconds in the training data but ranged from 1.5 to 10 seconds in the test data. The inductive bias of the unsupervised VAE made the representation useful because it was sensitive to changes in the input signals and detected the anomalies with high sensitivity when the operational borders were set to produce a false positive rate of 5 percent. At this threshold,

TABLE II: Evaluation of anomaly detection performance on four different datasets.

Dataset	AUC		Recall ($B = 0.95$)		Specificity ($B = 0.95$)		Odds Ratio ($B = 0.95$)	
	One-batch calibration	Dynamic calibration	One-batch calibration	Dynamic calibration	One-batch calibration	Dynamic calibration	One-batch calibration	Dynamic calibration
D1	0.9039	0.9974	1.0	1.0	0.2266	0.8729	1.4002	5.8545
D2	0.9620	0.9974	1.0	1.0	0.3146	0.8593	1.5769	6.9487
D3	0.9980	0.9980	1.0	1.0	0.6823	0.8614	7.4856	3.1971
D4	0.9425	0.9998	1.0	1.0	0.2245	0.8809	1.4151	7.7862
Avg	0.9516	0.9982	1.0	1.0	0.3620	0.8686	2.9695	5.9466

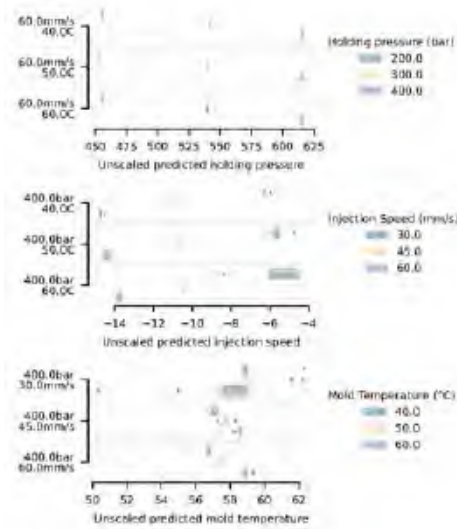


Fig. 5: Predicted machine parameters of a subset of D5. Each triplet of boxplots has 2 machine parameters fixed as indicated in the tick labels of the horizontal axis. The aim of the modeling is that the mini-batches produce correctly ordered predicted parameters $m_1 < m_2 < m_3$ or vice versa for correspondingly ordered set machine parameters. This is clearly achieved for holding pressure and injection speed. The scale and sign of the unscaled predictions are arbitrary.

the specificity was much lower than the expected percentage. By continuously re-calibrating the distribution parameters, the specificity increased above 80% but never reached the expected 95%. This is indicative of the data exhibiting slow drifts and that our assumption about the normality of the latent variables does not fully hold. We also observed that the estimated variance is not constant. In D3, for example, the variance of the first calibration batch was larger than throughout the remaining cycles. Here, the dynamic calibration reduced precision, triggering more false alarms.

Holding pressure and injection speed were proportionally predicted, yet the model did not fully disentangle them. This is not surprising as both parameters lead to a higher pressure peak. Changes in the mold temperature caused relatively little change in the pressure curves and therefore the system was

not sensitive to those changes.

Limitations. Although the test set consisted of six batches (datasets), it only represents a small fraction of possible production data. A wider study is needed to explore the limits of the method in terms of generalization performance.

Some molds produce multiple parts and host one pressure sensor for each part. The system evaluation in this paper is limited to analyzing each cavity in isolation. This ignores potential information from interactions between channels. For example, when a single cavity is clogged, this would affect mainly the signals of that cavity. If the inlet were to be clogged, the pressure reading of all the cavities would be affected, triggering anomalies across all channels. Such scenarios would require updated training and/or infusion of domain expert knowledge at the level of root cause detection.

Another limitation is that when the reference cycles are very similar, the variance of the distribution is underestimated. This reduced the specificity. Currently, the model can adapt to slow data drift. However, the industrial process still needs to be supervised. When data constantly drifts in an unexpected direction, leading to an anomaly, the operator should intervene.

Conclusion and outlook. This study demonstrates progress of practical importance in the automation of injection molding process monitoring. The anomaly detection is sensitive and reliable if the process is stable. The dynamic calibration improves the reliability if there is a slow drift in the representation. The estimated distribution generally underestimates the variance which leads to a specificity that is lower than if the observed data would follow the theoretical model.

The proposed system can be improved in several ways without changing the architecture. If controlling the false positive rate is critical, the density estimation must be improved in future work. This could be achieved by either finding a distribution with few parameters (such as the Student distribution) that fits the observations better or fitting a more complex distribution such as a Gaussian mixture [36]. The latter case would require more training data, and in both cases, the appeal of easy-to-interpret elliptical process boundaries would be lost. Enlarging the training data set would help make the network generalize better. Readily available process data without machine parameters or quality labels could be used for pre-training. However, more variability in the data may entail the need to increase the network capacity by adding depth or breadth to the network. In our experiments, we found that insufficient capacity caused the network to get trapped in local

minima where the posterior collapses on the prior [37].

The proposed system underestimated the variance as indicated by a specificity lower than 0.95 at an anomaly threshold corresponding to 0.05. The posterior variance σ of the VAE may be leveraged to estimate the batch-specific posterior better than with the distribution of the posterior mean μ alone.

Improving the disentanglement of the latent factors [38] is necessary to improve the system's utility further and could be achieved without a fundamental change in the architecture. On the other hand, transformer-based self-supervised models with dimensionality reduction capabilities [39], [40] may produce more accurate reconstructions and, therefore, a more expressive latent representation.

REFERENCES

- [1] M. Czepiel, M. Bańkosz, and A. Sobczak-Kupiec, "Advanced injection molding methods: Review," *Materials*, vol. 16, no. 17, 2023.
- [2] S. Farahani, N. Brown, J. Loftis, C. Krick, F. Pichl, R. Vaculik, and S. Pilla, "Evaluation of in-mold sensors and machine data towards enhancing product quality and process monitoring via industry 4.0," *Int J Adv Manuf Technol*, vol. 105, pp. 1371–1389, 2019.
- [3] T. Ageyeva, S. Horváth, and J. G. Kovács, "In-mold sensors for injection molding: On the way to industry 4.0," *Sensors*, vol. 19, no. 16, p. 3551, 2019.
- [4] V. Rousopoulou, A. Nizamis, T. Vafeiadis, D. Ioannidis, and D. Tzouvaras, "Predictive Maintenance for Injection Molding Machines Enabled by Cognitive Analytics for Industry 4.0," *Front Artif Intell*, vol. 3, 2020.
- [5] R. Olsen, "Ausbildung: Interesse an Lehrberufen im Kunststoffbereich nimmt weiter ab," *Kunststoff Information*, Bad Homburg. Available online: <https://www.kiweb.de/Default.aspx?pageid=199&docid=252599>, 2023.
- [6] M. Javaid, A. Haleem, R. P. Singh, and R. Suman, "Artificial intelligence applications for industry 4.0: A literature-based study," *J Ind Integr*, vol. 7, no. 01, pp. 83–111, 2022.
- [7] L. Waltersmann, S. Kiemel, J. Stuhlsatz, A. Sauer, and R. Mieke, "Artificial intelligence applications for increasing resource efficiency in manufacturing companies—a comprehensive review," *Sustainability*, vol. 13, no. 12, p. 6689, 2021.
- [8] M. C. Le, S. Belhabib, C. Nicolazo, P. Vachot, P. Mousseau, A. Sarda, and R. Deterre, "Pressure influence on crystallization kinetics during injection molding," *J Mater Process Technol*, vol. 211, no. 11, pp. 1757–1763, 2011.
- [9] D. C. Angstadt and J. P. Coulter, "Cavity pressure and part quality in the injection molding process," in *Proc. ASME IMECE*, vol. 16622. American Society of Mechanical Engineers, 1999, pp. 7–17.
- [10] "Process monitoring and control system ComoNeo for injection molding," Available online: <https://www.kistler.com/DE/en/cp/process-monitoring-and-control-system-comoneo-5887a/P0000382>, 2023.
- [11] P. Waibel, S. Haltmeier, R. Vaculik, T. Wuhmann, and N. Pascher, "Cavity pressure-based machine learning service for advanced injection molding processes," White paper, available online: <https://www.kistler.com/INT/en/monitoring-and-control-of-injection-molding-processes/C00000041>, 2021.
- [12] T. Stadelmann, T. Klamt, and P. H. Merkt, "Data centrism and the core of data science as a scientific discipline," *Archives of Data Science, Series A*, vol. 8, no. 2, 2022.
- [13] P.-P. Luley, J. M. Deriu, P. Yan, G. A. Schatte, and T. Stadelmann, "From concept to implementation: The data-centric development process for AI in industry," in *Proc. SDS*, 2023, pp. 73–76.
- [14] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," *Stat*, vol. 1050, p. 1, 2014.
- [15] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "Beta-VAE: Learning basic visual concepts with a constrained variational framework," in *Proc. ICLR*, 2016.
- [16] T. Stadelmann, M. Amirian, I. Arabaci, M. Arnold, G. F. Duivesteijn, I. Elezi, M. Geiger, S. Lörrwald, B. B. Meier, K. Rombach *et al.*, "Deep learning in the wild," in *Proc. ANNP*, 2018, pp. 17–38.
- [17] V. Ketonen and J. O. Blech, "Anomaly detection for injection molding using probabilistic deep learning," in *Proc. ICPS*, 2021, pp. 70–77.
- [18] J. Gim and B. Rhee, "Novel analysis methodology of cavity pressure profiles in injection-molding processes using interpretation of machine learning model," *Polymers*, vol. 13, no. 19, p. 3297, Jan. 2021.
- [19] P. Yan, A. Abdulkadir, P.-P. Luley, M. Rosenthal, G. A. Schatte, B. F. Grewe, and T. Stadelmann, "A comprehensive survey of deep transfer learning for anomaly detection in industrial time series: Methods, applications, and directions," *IEEE Access*, 2024.
- [20] T. Stadelmann, V. Tolkachev, B. Sick, J. Stampfli, and O. Dürr, "Beyond ImageNet: deep learning in industrial practice," *Applied data science: lessons learned for the data-driven business*, pp. 205–232, 2019.
- [21] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep learning for anomaly detection: A review," *ACM CSUR*, vol. 54, no. 2, pp. 1–38, 2021.
- [22] L. Tugener, J. Schmidhuber, and T. Stadelmann, "Is it enough to optimize CNN architectures on ImageNet?" *Front Comput Sci*, vol. 4, p. 1041703, 2022.
- [23] B. Maschler, H. Vietz, H. Tercan, C. Bitter, T. Meisen, and M. Weyrich, "Insights and example use cases on industrial transfer learning," *Proc. CIRP*, vol. 107, pp. 511–516, 2022.
- [24] C. H. Tan, V. C. Lee, M. Salehi, S. Marusic, S. Jayawardena, and D. Lucke, "A fully unsupervised and efficient anomaly detection approach with drift detection capability," in *Proc. ICDMW*, 2021, pp. 312–321.
- [25] D. Wu, D. Zhou, and M. Chen, "Probabilistic stationary subspace analysis for monitoring nonstationary industrial processes with uncertainty," *IEEE TII*, vol. 18, no. 5, pp. 3114–3125, 2022.
- [26] S. Saurav, P. Malhotra, V. TV, N. Gugulothu, L. Vig, P. Agarwal, and G. Shroff, "Online anomaly detection with concept drift adaptation using recurrent neural networks," in *Proc. CODS-COMAD*, 2018, pp. 78–87.
- [27] Z. Yang, I. Soltani, and E. Darve, "Anomaly detection with domain adaptation," in *Proc. CVPR*, 2023, pp. 2957–2966.
- [28] Y. Lockner, C. Hopmann, and W. Zhao, "Transfer learning with artificial neural networks between injection molding processes and different polymer materials," *J Manuf Process*, vol. 73, pp. 395–408, 2022.
- [29] H. Tercan, A. Guajardo, and T. Meisen, "Industrial transfer learning: Boosting machine learning in production," in *Proc. INDIN*, vol. 1, 2019, pp. 274–279.
- [30] T. Chen, V. Sampath, M. C. May, S. Shan, O. J. Jorg, J. J. Aguilar Martín, F. Stamer, G. Fantoni, G. Tosello, and M. Calao, "Machine learning in manufacturing towards Industry 4.0: From 'for now' to 'four-know'," *Applied Sciences*, vol. 13, no. 3, 2023.
- [31] Z.-H. Wang, F.-C. Wen, Y.-T. Li, and H.-H. Tsou, "A novel sensing feature extraction based on mold temperature and melt pressure for plastic injection molding quality assessment," *IEEE Sens. J.*, vol. 23, no. 7, pp. 7451–7459, 2023.
- [32] N. Simmler, P. Sager, P. Andermatt, R. Chavarriaga, F.-P. Schilling, M. Rosenthal, and T. Stadelmann, "A survey of un-, weakly-, and semi-supervised learning methods for noisy, missing and partial labels in industrial vision applications," in *Proc. SDS*, 2021, pp. 26–31.
- [33] P. C. Mahalanobis, "On the generalized distance in statistics," *Proceedings of the National Institute of Sciences of India*, pp. 49–55, 1936.
- [34] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, San Diego, CA, USA, 2015.
- [35] Schatte, Gerrit A. and Abdulkadir, Ahmed and Yan Peng and Rosenthal, Matthias and Schwizer, Simone and Wildmann, Damian and Vaculik, Robert and Aguzzi, Giulia and Schirmer, Mathias and Thilo Stadelmann, "Verfahren und Vorrichtung zur Überwachung eines zyklischen Herstellprozesses," Swiss Patent Request EP23 212 020, Nov 4, 2023.
- [36] T. Stadelmann and B. Freisleben, "Dimension-decoupled Gaussian mixture model for short utterance speaker recognition," in *Proc. ICPR*, 2010, pp. 1602–1605.
- [37] Y. Wang, D. Blei, and J. P. Cunningham, "Posterior collapse and latent variable non-identifiability," *Proc. NeurIPS*, vol. 34, pp. 5443–5455, 2021.
- [38] S. Hahn and H. Choi, "Disentangling latent factors of variational auto-encoder with whitening," in *Proc. ICANN*. Springer, 2019, pp. 590–603.
- [39] R. Ran, T. Gao, and B. Fang, "Transformer-based dimensionality reduction," *arXiv:2210.08288*, 2022.
- [40] B. Heo, S. Yun, D. Han, S. Chun, J. Choe, and S. J. Oh, "Rethinking spatial dimensions of vision transformers," in *Proc. ICCV*, October 2021, pp. 11 936–11 945.

Originally published in Proceedings of the 11th IEEE Swiss Conference on Data Science (SDS'24), Zurich, Switzerland, May 31, 2024, DOI 10.1109/SDS60720.2024.00036.

So you want your private LLM at home? A survey and benchmark of methods for efficient GPTs

Lukas Tuggener^{1,2,3}, Pascal Sager^{1,5,7}, Yassine Taoudi-Benchekroun^{6,7}, Benjamin F. Grewe^{5,6,7}, and Thilo Stadelmann^{1,4,5}

¹ ZHAW CAI, Winterthur, Switzerland ² USI, Lugano, Switzerland ³ RWA AG, Winterthur, Switzerland

⁴ ECLT, Venice, Italy ⁵ AlpineAI AG, Davos, Switzerland ⁶ ETH AI Center, Zurich, Switzerland

⁷ ETH/UZH INI, Zurich, Switzerland

{tugg, sage, stdm}@zhaw.ch, ytaoudi@ethz.ch, bgrewe@ethz.ch

Abstract—At least since the introduction of ChatGPT, the abilities of generative large language models (LLMs), sometimes called GPTs, are at the center of the attention of AI researchers, entrepreneurs, and others. However, for many applications, it is not possible to call an existing LLM service via an API due to data protection concerns or when no task-appropriate LLM exists. On the other hand, deploying or training a private LLM is often prohibitively computationally expensive. In this paper, we give an overview of the most important recent methodologies that help reduce the computational footprint of LLMs. We further present extensive benchmarks for seven methods from two of the most important areas of recent progress: model quantization and low-rank adapters, showcasing how it is possible to leverage state-of-the-art LLMs with limited resources. Our benchmarks include resource consumption metrics (e.g. GPU memory usage), a state-of-the-art quantitative performance evaluation as well as a qualitative performance study conducted by eight individual human raters. Our evaluations show that quantization has a profound effect on GPU memory requirements. However, we also show that these quantization methods, contrary to how they are advertised, cause a noticeable loss in text quality. We further show that low-rank adapters allow effective model fine-tuning with moderate compute resources. For methods that require less than 16 GB of GPU memory, we provide easy-to-use Jupyter notebooks that allow anyone to deploy and fine-tune state-of-the-art LLMs on the Google Colab free tier within minutes without any prior experience or infrastructure.

Index Terms—Large Language Models, LlamaV2, fine-tuning, LLM quantization, LLM deployment

I. INTRODUCTION

Recent Large Language Models (LLMs) have shown remarkable performance on a wide range of language tasks [1], which has earned them an impressive rise in interest [2] that has extended beyond the typical aficionados in academia and industry. Impressive results have been achieved using little prior knowledge and purely leveraging data [3] thanks to the adoption of the transformer architecture [4] (a.k.a. Fast Weight Programmers [5]), which enables each layer of an LLM to focus on specific parts of the input sequence depending on the need. Effectively, models give varying degrees of importance to different elements in the input sequence dynamically, depending on their relevance to the current step in the sequence generation process. This modeling power comes at a cost of crucial computational complexity—transformers

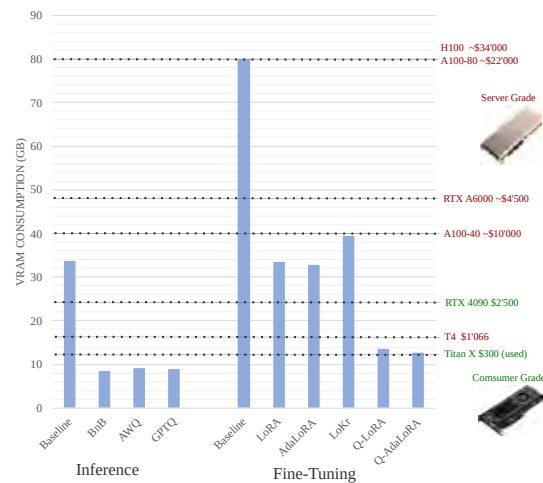


Fig. 1. The VRAM consumption of all benchmarked methods. See Section III-B for inference and Section III-C for fine-tuning. For context, we illustrate the VRAM capacities and list approximate prices of the most popular GPUs in the AI space. Note that GPU prices are very volatile and time dependent.

scale quadratically with sequence length ($\mathcal{O}(n^2d)$, with n the sequence length, d embedding dimension). However, language transformers only unlock their full potential when having billions of trainable parameters (i.e. putting the L into LLM) [6]. Combined with their computational scaling properties, this makes LLMs notoriously difficult and costly to both train and deploy. Training models for custom applications from scratch requires such large amounts of data and large model sizes that it is only possible for large corporations and universities with substantial resources. Nonetheless, there has been a trend among some of the tech giants to open source even their most capable models, most notably LlamaV2 [7] by Meta. The open availability of these models has enabled a whole new research field concerned with methods that allow the efficient use of pre-trained LLMs on modest hardware configurations.

In this paper, we give an overview of these new technologies

by surveying the state of the art. For the two most important among them, quantization (see Section II-A1) and low-rank approximators (see Section II-A2), we present novel benchmark results (see Fig. 1). We evaluate three quantized inference methods and four low-rank approximators for fine-tuning. We benchmark these methods twofold: We first investigate the computational properties of each method, reporting resource consumption and computation speed. In the second step, we investigate the quality of the model outputs by using automated evaluation metrics and conducting a qualitative study on select outputs. The GPU memory (VRAM) requirements of some of the tested methods are low enough that they can be deployed on the free tier of a Google Colab instance. For these methods, we provide easy-to-understand Jupyter notebooks that enable anyone to deploy a state-of-the-art LLM within minutes without any prior knowledge or infrastructure.

The remainder of this paper is structured as follows: in Section II, we give an overview of the key technologies that enable LLMs on limited hardware; in Section III-A, we introduce our benchmark methodology; Sections III-B and III-C contain the benchmark results for model inference and fine-tuning, respectively. In Section III-D, we share our Colab notebooks before we draw conclusions in Section IV.

II. SURVEY OF KEY TECHNOLOGIES

In this section, we present an overview of the innovations that have enabled model fine-tuning, inference, and performance evaluation of LLMs on a moderate compute budget. We omit to survey how LLMs should be trained from scratch as this undertaking remains extremely costly, being out of scope for most organizations and practical applications. For general design patterns of efficient deep learning, see [8], [9].

A. Efficient LLM Execution and Adaptation

1) *Quantization*: The biggest hurdle to the wide adoption of LLMs is their VRAM requirements. The parameters and activation states of these models are usually stored in a 32- or sometimes 16-bit format. It has been shown that quantization methods that can properly represent the characteristic emergent structures in transformer [4] weights can reduce the parameter and activation formats to an 8-bit representation with minimal loss of arithmetic accuracy [10]. Recently, a 4-bit NormalFloat data type has been introduced that is theoretically optimal for storing normally distributed weights [11].

When quantizing a fully pre-trained transformer model, the quantization can take the explicit structure of the model weights into account, making quantization even more efficient: GPTQ [12] leverages approximate second-order information to generate highly accurate approximations with 3 to 4 bits per weight. Activation-aware Weight Quantization (AWQ) [13] achieves a decrease in quantization error by deriving the importance of each weight through observing their activations before quantization, then protecting the 1% most important weights from being quantized.

When also training a model (not only running inference), the required VRAM is drastically increased due to the variables

required by the optimizer. A straightforward way to address this is to also quantize the optimizer's weights [14]. Quantization is very widely adopted and considered to be the most effective method for reducing VRAM consumption.

2) *Low-Rank Approximators*: While quantization methods are often sufficient to run LLM inference on constrained hardware, VRAM constraints prevent fine-tuning LLMs on a single GPU even when using optimizer quantization. Low-rank adaptation of large language models (LoRA) [15] offers an alternative: LoRA freezes the pre-trained model weights and injects trainable low-rank decomposition matrices into each layer of the Transformer. This way the base network can be run in "inference mode" and only the low-rank matrices are trained as a differential update to the base network. AdaLoRa [16] develops this concept further and assigns parameter budgets to different weight matrices according to an importance metric. LoKr [17] recognizes limited representation power of above low-rank representations and suggests using the Kronecker product instead. Low-rank approximators are the key innovation that democratized LLM fine-tuning. In resource-constrained settings they are mandatory.

3) *Pruning*: A popular strategy to alleviate the high inference cost of deep neural networks is "model pruning" [18]–[20]: Starting from a pre-trained network, one can iteratively remove redundant parameters and retrain the model while maintaining a test accuracy comparable to the original full-sized network. While Large Language Models are natural candidates for network pruning techniques, they have received little interest from the community compared to the other popular compression techniques such as quantization (see II-A1). A likely explanation for this is that standard pruning methodologies require retraining the network, which is too costly in the case of LLMs. Recent works have however shown promising results with pruning without relying on retraining networks [21], [22].

4) *CPU Parameter Offloading*: CPU offloading in training deep neural networks involves transferring some computations from the GPU to the CPU to alleviate memory constraints and improve training efficiency. Optimal strategies for offloading activations, gradients, and model parameters to the CPU ultimately enable the training of larger models. Microsoft has been at the forefront of this line of research and introduced the open-source library DeepSpeed [23] as well as ZeRO [24]. This includes a "ZeRO-Offload" feature, which performs optimized CPU parameter offloading, enabling large models with up to 13 billion parameters to be efficiently trained on a single GPU. Smart load sharing between CPUs and GPUs is very important in large multi-GPU or even multi-cluster operations. When working with a single GPU it is usually preferable to compute everything on that device to avoid costly data transfers.

5) *Stateless Optimizers*: Numerous prevalent optimization algorithms come with the cost of additional parameters. For instance, Adam [25] and AdamW [26] involve the computation of both the moving average of first-order moments (mean of gradients) and second-order moments (uncentered variance of gradients) for each parameter. Consequently, they incur the

cost of tripling the parameter count. An alternative to quantizing the optimizer weights (see Sec. II-A1) is to use a stateless optimizer like stochastic gradient descent. The drawback of plain SGD is its typically inferior convergence properties. However, this limitation can be mitigated by introducing a learning rate scheduler [27] that systematically reduces the learning rate, ensuring the attainment of a local minimum. Running a stateless optimizer is an efficient way of reducing VRAM requirements. The incurred loss in convergence speed depends on the problem at hand and it is best to investigate in a preliminary study if it is a worthwhile tradeoff.

6) *Gradient Checkpointing*: Conventionally, intermediate activations from the forward pass are stored in the VRAM for gradient computation during backpropagation, but this becomes impractical in scenarios with limited memory or very large models. Gradient (or: activation) checkpointing [28], [29] addresses this challenge by selectively caching a subset of crucial intermediate activations for gradient computation, dynamically recomputing the remaining activations during the backward pass. This turns out to make a better trade-off than re-computing all activations: The flexibility of gradient checkpointing in choosing activations for checkpointing allows tailored optimization based on a model's specific memory and computational demands. Gradient checkpointing can help to further reduce memory consumption when reducing the batch size is no longer possible. It comes, however, at a steep cost in training speed.

7) *Gradient Accumulation*: Gradient accumulation [30] is a technique that allows training on large "virtual batches" that are bigger than the maximum batch size fitting into the available VRAM. This is achieved by accumulating the gradients over multiple batches and only running an optimizer step once enough gradients for the desired "virtual batch size" have been collected. This is often necessary when working with a limited VRAM budget. Gradient accumulation ensures that the necessary batch size can be reached which reduces the noise in the gradient updates enough for the LLMs to properly converge.

8) *Software Engineering*: LLMs typically operate in an autoregressive fashion [4], predicting the succeeding token based on the prior sequence. Consequently, output generation time increases as the predicted sequence's length grows. In the context of token streaming (as, e.g., in ChatGPT's user interface), the output can be exhibited token by token, as the already produced sequence is directly presented to the user. Furthermore, the autoregressive approach also facilitates continuous batching [31]. In this method, incoming requests are accumulated in a queue, and after each autoregressive step, they are incorporated into the currently processed batch, provided the memory limit permits. Furthermore, sequences that reach the end-of-sequence token are removed from the batch to free up space, allowing other sequences in the queue to be processed.

These measures are crucial when running LLMs in production; Token streaming should be used when the users are actively waiting for the response, allowing users to promptly

read the result, even if it hasn't been fully generated. Continuous batching, on the other hand, is particularly beneficial in scenarios where the timing and quantity of requests are uncertain, for example, in multi-user settings.

B. Evaluation

Various metrics such as perplexity [32], BLEU score [33], and F1 score have been proposed to gauge the performance of language models on specific tasks, including translation [34], and text classification [35]. However, LLM-based assistants have started to exhibit more general capabilities across diverse tasks such as writing, chatting, and coding [7], [36]–[38], making evaluating their broad capabilities more challenging. Existing benchmarks to evaluate LLMs include core-knowledge benchmarks [39]–[45] that typically require LLMs to generate a short, specific answer to benchmark questions that can be automatically validated. Instruction-following benchmarks [46]–[50] use more open-ended questions and more diverse tasks to evaluate LLMs after instruction fine-tuning. Conversational benchmarks [51]–[53] focus on dialogue; however, the diversity and complexity of their questions often fall short in challenging the capabilities of the latest models.

Many of these performance metrics primarily center on assessing multi-turn dialogues involving open-ended inquiries, or they fail to pose a sufficient challenge for contemporary cutting-edge models. Although human preferences directly gauge a chatbot's efficacy in multi-turn interactions between humans and AI, this approach lacks scalability. Based on this insight, Zheng et al. [54] propose to bridge this gap by using the novel MT-bench benchmark that tests the core capabilities of state-of-the-art models while leveraging an auxiliary LLM model as a judge. Remarkably, their findings reveal that auxiliary LLM models exhibit a level of agreement comparable to that of human annotators, rendering them suitable for evaluating other LLMs.

In the next section, we apply MT-bench in conjunction with an auxiliary (LLM) to assess the efficacy of a model across diverse tasks. Moreover, this metric proves apt for discerning whether model performance drops during fine-tuning when using, for example, quantization methods, as it not only provides a score per category but also allows pairwise comparison between models.

III. BENCHMARKING PERFORMANCE AND QUALITY

We present benchmarks for two distinct use cases: First, we assess model inference by employing a pre-trained model. Second, we examine fine-tuning on custom data. Model quantization and LoRA are the most promising, impactful, and universally applicable among the plethora of techniques introduced above. Therefore, we chose them for benchmarking. In both scenarios, we conduct a comparative analysis between a given model and the same model incorporating additional quantization techniques. In the inference scenario, we benchmark quantization techniques, while in the fine-tuning scenario, we test multiple low-rank approximations. One focus lies on evaluating VRAM usage, speed, and model

performance, as these metrics are pivotal for the effective deployment of models in resource-constrained environments. The second focus lies on gauging the quality of the generated conversations. Since it is the most capable openly available LLM, we use LlamaV2 [7] as the basis for all of our experiments. We use its 7-billion parameter variant – larger versions require a substantial GPU infrastructure, even when paired with the techniques introduced above.

A. Benchmarking methodology

1) *Resource usage and speed metrics*: For inference, we systematically report metrics acquired during the quantitative evaluation (c.f. Section III-A2) on the MT benchmark. This includes VRAM usage (the most constraining factor for the seamless execution of state-of-the-art models with billions of parameters), generated tokens per second, and overall power consumption in watt-hours (Wh). For fine-tuning, we provide metrics obtained during the re-training, including VRAM usage, processed batches per second, and overall power consumption. For simplicity, we constrain the models to a batch size of 1, even though the aforementioned methods might reduce VRAM requirements to allow an increase in batch size, contingent on available computational resources.

The experiments are conducted using a single Nvidia A-100 40GB GPU. The selection of this GPU was based on its ample VRAM, allowing the execution of all model versions (except direct fine-tuning which is out of the scope of this work), both with and without quantization, facilitating a fair comparison. We also report results generated using a Nvidia T4 16GB GPU to demonstrate the feasibility of running and fine-tuning models on constrained hardware (“at home”). It is important to note that, due to the memory limitations of the Nvidia T4 16GB GPU, only models employing quantization methods could be executed on it.

2) *Quantitative Evaluation*: We employ the MT-benchmark that contains 80 high-quality multi-turn questions described in [54]. They are manually designed to evaluate the models’ proficiency in multi-turn conversation and instruction-following capabilities. The questions are organized into eight categories, each holding ten challenging questions: writing, roleplay, extraction, reasoning, math, coding, STEM knowledge, and humanities/social science knowledge. Each multi-turn question comprises an initial plus a subsequent inquiry.

The multi-turn questions are fed into all model versions, and the resulting responses are stored. Subsequently, OpenAI’s GPT-4 model is utilized to assess response quality by assigning a score ranging from 1 to 10 to each answer. To obtain the score, we follow the prompting scheme of Zheng et al. [54], asking GPT-4 to evaluate the model’s response for various factors such as “relevance”, “accuracy”, and “depth” before providing an overall score. The average score per category is then computed to provide a comprehensive evaluation of the model’s performance across different categories. In addition to this single-answer evaluation, a pairwise comparison is conducted between the baseline model and a second model version that has been tuned for efficiency. In this process, the

responses from both models are presented to the auxiliary LLM, GPT-4, which is then tasked with determining the superior model or declaring a tie. Both procedures adhere to the methodology proposed by Zheng et al. [54].

3) *Qualitative Evaluation*: As automatic evaluation of generative language models is still in its infancy, it is important to also qualitatively inspect outputs to assess the quality of a model. We systematically do this by prompting our models with the same catalog of five questions sourced from MT-bench. Then, eight individual human judges rate all the outputs on a scale from 1 (useless) to 5 (perfect answer). Finally, we average across the marks given by each judge to obtain an objective score. We use the following five multi-turn questions to assess the ability of the models on a broad range of topics:

Q1 (Writing) Round 1: *Compose an engaging travel blog post about a recent trip to Hawaii, highlighting cultural experiences and must-see attractions.*

Round 2: *Rewrite your previous response. Start every sentence with the letter A.*

Q2 (Reasoning) Round 1: *Thomas is very healthy, but he has to go to the hospital every day. What could be the reasons?*

Round 2: *Can you explain why the above question is interesting?*

Q3 (Math) Round 1: *Benjamin went to a bookstore and purchased a variety of books. He bought 5 copies of a sci-fi novel, each priced at \$20, 3 copies of a history book priced at \$30 each, and 2 copies of a philosophy book for \$45 each. What was the total cost of his purchases?*

Round 2: *Suppose Benjamin decides to sell each of these books at a 25% markup from the price he purchased them. What would be his total revenue if he sold all the books he bought?*

Q4 (Stem) Round 1: *Photosynthesis is a vital process for life on Earth. Could you outline the two main stages of photosynthesis, including where they take place within the chloroplast, and the primary inputs and outputs for each stage?*

Round 2: *How much energy can a tree produce through photosynthesis in its lifetime? Please provide an estimate using actual numerical values and thoroughly explain your thought process step-by-step.*

Q5 (Humanities) Round 1: *Which methods did Socrates employ to challenge the prevailing thoughts of his time?*

Round 2: *Let’s bring Socrates to the modern world. Generate a conversation between Socrates and Bill Gates to debate on generative AI for education.*

B. Results for Efficient Inference

In the subsequent analysis, we assess different quantization techniques with a focus on computational, quantitative, and qualitative performance. Out of the numerous open-source LLMs [1], we selected “LlamaV2-7B” and its dialogue-fine-tuned version “LlamaV2-7B-Chat” as the basis for this evaluation. This choice is based on the models’ good performance with a manageable, popular parameter count of “just” 7B [7].

Resource usage and speed metrics. Table I presents an overview of the computational requirements of the LlamaV2-7B-Chat model and its variants employing quantization techniques during the processing of the MT-bench dataset. The first column shows the VRAM usage. It is important to acknowledge that VRAM utilization may vary based on the sampled model response that is reused as conversational context in subsequent follow-up questions. The adoption of

TABLE I
VRAM USAGE, TOKENS PROCESSED PER SECOND, AND THE OVERALL POWER CONSUMPTION DURING INFERENCE ON MT-BENCH. THE MODEL USED IS LLAMA V2-CHAT, OPTIONALLY COMBINED WITH BITS-AND-BYTES (BNB), NESTED BNB, AWQ, AND GPTQ QUANTIZATION TECHNIQUES.

GPU	VRAM usage [GB]		avg. tokens/s		power consumption [Wh]	
	A100	A100	T4	A100	T4	
LlamaV2-7B-Chat	33.80	36.06	out of memory	92.39	out of memory	
LlamaV2-7B-Chat BnB	8.40	25.45	16.07	63.53	57.09	
LlamaV2-7B-Chat AWQ	9.10	86.66	26.32	33.72	38.96	
LlamaV2-7B-Chat GPTQ	9.02	34.20	28.33	52.56	32.28	

quantization techniques enables the execution of the model on GPUs with constrained VRAM, such as the Nvidia T4, as the VRAM requirements reduce from 33.80GB to below 10GB. Alternatively, in scenarios where sufficient memory is available, this facilitates the utilization of batch sizes that are three to four times larger, thereby enabling the concurrent processing of a higher volume of requests.

When examining the tokens processed per second using the A-100 GPU, it becomes evident that quantization via BnB (25.45 tokens/s) exhibits significantly slower performance compared to the baseline without quantization (36.06 tokens/s). Thus, squeezing parameters and activations into lower precision datatypes reduces memory requirements on the cost of speed. In contrast, more advanced algorithms such as GPTQ and AWQ can even preserve or improve the throughput while having a reduced memory footprint: GPTQ quantization achieves a throughput of 34.2 tokens/s, nearly matching the baseline. Remarkably, AWQ quantization attains an impressive throughput of 86.66 tokens/s, approximately 2.4 times faster than the baseline. Unfortunately, this enhanced performance does not scale to the T4 GPU due to this GPU's reliance on the older Turing architecture. The T4 GPU lacks the novel CUDA kernels of the Ampere architecture, which is extensively utilized by AWQ. Consequently, AWQ on the T4 GPU achieves performance similar to GPTQ. However, as previously elucidated, the quantization methods enable larger batch sizes and the concurrent processing of more tokens, thereby enhancing overall throughput. Thus, even slower methods like BnB could outperform the baseline when quantization is employed to increase the batch size.

Quantization techniques also allow for substantially diminishing power consumption. This is attributed to the utilization of lower-precision datatypes for representing model parameters and activations, resulting in enhanced computational efficiency. On the A-100 GPU, the standard model consumes 92.39Wh, whereas BnB has a power consumption of 63.53Wh, and nested BnB consumes 69.15Wh. GPTQ and AWQ employ similar datatypes as BnB, but due to their higher throughput, these methods achieve an even lower total power consumption of 33.72Wh (AWQ) and 52.56Wh (GPTQ), respectively. Unfortunately, due to the aforementioned reasons, AWQ struggles to transfer its high efficiency to the older GPU.

To summarise, from a purely computational perspective, AWQ quantization is preferable if the GPU is equipped with compatible CUDA drivers; otherwise, GPTQ is recommended. These approaches demand significantly less VRAM compared

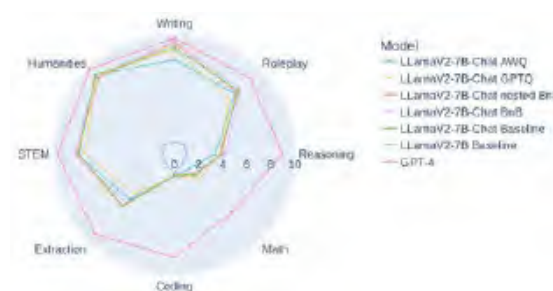


Fig. 2. Single score evaluation results for LlamaV2-7B-Chat with different quantization techniques and the state-of-the-art GPT-4 for comparison.

to the baseline model while maintaining or increasing processing speed, even when the batch size remains unchanged. Unfortunately, AWQ and GPTQ cannot be used for model training, making BnB the preferred method for quantization while learning model weights.

Quantitative evaluation. To quantitatively assess the influence of quantization on LlamaV2-7B-Chat's output quality, we utilize the single-score judgment based on MT-bench. Additionally, we include evaluations of LlamaV2-7B as a lower bound and GPT-4 as an upper bound. Please note that GPT-4, when used as a judge, favors its own answers with a 10% bias in pair-comparisons, suggesting that it rates itself also slightly better than other models in single answer grading [54]. LlamaV2-7B is used as the lower bound due to its identical architecture but lack of fine-tuning for dialogue, a requirement for MT-bench. The findings are presented in Figure 2. As expected, the LlamaV2-7B-Chat model and its quantized versions are noticeably better than the lower bound and worse than the upper bound. There is only a marginal performance difference between the baseline and the quantized models, suggesting that the application of quantization techniques neither degrades nor enhances performance.

Qualitative evaluation. Table II shows the average grade for the answers of all the quantized versions of LlamaV2-Chat to our five selected questions (see Section III-A3). Notably, the non-quantized model achieves the highest average grade of 3.35, surpassing GPTQ with an average grade of 3.1 and BnB with an average grade of 2.9. In contrast, AWQ received a lower average rating of 1.95 compared to the other quantization methods. This discrepancy suggests that humans

TABLE II
AVERAGE (ACROSS 8 HUMAN RATERS) USEFULNESS EVALUATION
(1=USELESS, 5=PERFECT) OF ANSWERS FROM QUANTIZED VERSIONS OF
LLAMA V2-CHAT

	Q1	Q2	Q3	Q4	Q5	Average
no quantization	3.75	1.25	4.5	4	3.25	3.35
BnB	2.25	2	2.75	3.5	4	2.9
AWQ	1.5	1.5	1.5	2.5	2.75	1.95
GPTQ	1.25	1.75	4.75	3.75	4	3.1

do not favor AWQ as a quantization technique, which is in contrast to the outcomes of the quantitative analysis, where no performance gap could be observed. Our results indicate that quantization techniques improve computational efficiency but degrade the overall dialogue quality.

C. Results for Efficient Model Fine-Tuning

When there is no task-appropriate LLM available, it becomes necessary to fine-tune an LLM on a new dataset. To evaluate the effect of efficiency techniques for LLM fine-tuning, we fine-tune a LlamaV2 base model on the dataset that was designed for the training of Guancano [11] and that is a subset of the OpenAssistant [53] dataset. Thus, we train the model to behave like a chat-based assistant rather than a general generative model. While this is of no practical importance as a new model due to the existence of LlamaV2-Chat, it allows us to keep a consistent evaluation scheme across all experiments for this benchmarking. Fine-tuning a LlamaV2 model without a low-rank adapter is only possible with very large computational resources (at least 80GB VRAM on a single chip), therefore we consider the widely used LoRA [15] method as our baseline. Additionally, we fine-tune models using AdaLoRA [16] and LoKr [17]. We also test quantized versions of LoRA (Q-LoRA) and AdaLoRA (Q-AdaLoRA), which provide a further reduction in VRAM requirement. We train our models with a constant learning rate of 0.0001 and a weight decay of 0. We use a batch size of 1 and 8 gradient accumulation steps. We cap the gradient norm at 0.3 and train for 15,000 steps. LoKr has a relatively heavy VRAM load, therefore we have to reduce the maximum sequence length to 400 tokens instead of the default of 500, even on a 40GB GPU. To enable fair comparisons, we keep this training configuration (including max sequence length) consistent across all methods. The full training details can be found in the public WandB logs linked here: https://github.com/tuggeluk/LMM_at_home. We have opted for this relatively short fine-tuning due to our limited computational resources. This setup allows us to compare the different low-rank adaptation methods, but the resulting models are not expected to be competitive with the highly tuned LlamaV2-Chat.

Resource usage and speed metrics. Table III shows an overview of the resources consumed by the different fine-tuning methods for the A100 and T4 GPUs. Among the non-quantized adapters, LoKr requires a higher amount of memory than LoRA and AdaLoRA. Quantizing the adapter slashes the memory required for fine-tuning by almost two-thirds,

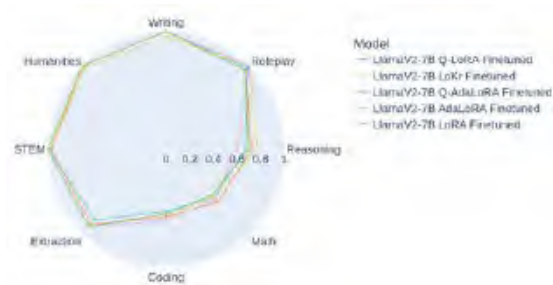


Fig. 3. Pair-wise comparison results of LlamaV2-7B vs. fine-tuned models.

allowing the use of T4 chips that only have 16GB of memory. The processing speed does not deviate too sharply between the methods with the fastest processing (LoRA: 0.217 steps/s) and the slowest processing (LoKr: 0.15 steps/s) on an A-100. Quantizing causes a slowdown of roughly 20% in both cases, comparable to the BnB quantization during inference. The T4 is much slower at 0.04 steps/s for both methods, yielding an increase in training time of more than four-fold. The overall power consumption on the A-100 is highly correlated with the training time, while the reduced memory usage of the quantized methods has a slim effect on the energy consumption, making the faster, unquantized training a more energy-efficient option. While the T4 has a much lower peak power consumption, this is largely compensated by longer training time, leading to similar energy usage for both GPU models.

Quantitative evaluation. To assess the impact of efficiency-optimized fine-tuning on chat quality, we conduct a pair-wise comparison using MT-bench. In this comparison, a judging model assigns scores of 1.0 if the fine-tuned version is superior to the baseline model, 0.5 if they perform equally, and 0.0 if the fine-tuned version is inferior. The results, depicted in Figure 3, reveal that our fine-tuned models consistently outperform the baseline, consisting of a stock LlamaV2, across all categories. This underscores the effectiveness of our fine-tuning. Notably, the fine-tuned models exhibit especially superior performance in the categories of writing, roleplay, extraction, STEM, and humanities, with average scores hovering around 1.0, indicating their consistent superiority over the baseline in nearly every instance.

In addition to conducting pairwise comparisons, we present outcomes based on single-score assessments using MT-bench. The outcomes are illustrated in Figure 4. The results from our fine-tuned models follow a similar pattern as the results from the quantized LlamaV2-Chat version used for inference (cf. Figure 2), i.e., work better in specific categories than others. However, overall, they perform worse than the models used during inference. This discrepancy arises from our use of a simplified and short fine-tuning approach, wherein we refrain from manually evaluating multiple checkpoints to identify the optimal model version. Nonetheless, this validation affirms

TABLE III
VRAM USAGE, TOKENS PROCESSED PER SECOND, AND OVERALL POWER CONSUMPTION DURING FINE-TUNING. THE BASE MODEL USED IS LLAMA V2-7B, FINE-TUNED WITH LOW-RANK APPROXIMATORS LoRA, AdaLoRA, LoKr, QUANTIZED LoRA, AND QUANTIZED AdaLoRA.

GPU	VRAM usage [GB]		steps/s	power usage [Wh]	
	A100	T4		A100	T4
LoRA	33.4	0.217	out of memory	6,641	out of memory
AdaLoRA	32.7	0.203	out of memory	6,560	out of memory
LoKr	39.5	0.15	out of memory	10,080	out of memory
Q-LoRA	13.5	0.178	0.04	7,990	7,402
Q-AdaLoRA	12.7	0.168	0.04	7,936	7,843

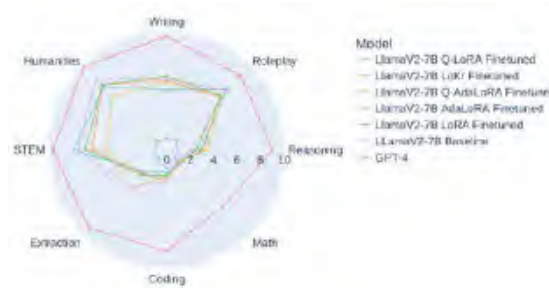


Fig. 4. Single score evaluation results for pre-trained LlamaV2-7B (baseline) and our fine-tuned versions.

that our results are comparable to the open-source fine-tuned versions.

When comparing the results from the fine-tuned versions, only subtle distinctions are evident. For instance, LoKr exhibits a slightly inferior performance in tasks related to humanities and STEM compared to the other models. Conversely, LoKr, along with AdaLoRA, surpasses other models in tasks involving information extraction. Moreover, quantization methods appear to have negligible effects on the overall quality of the model. The mean rating for LoRA stands at 4.58, and its quantized counterpart, Q-LoRA, is rated at 4.63 (+0.05). Similarly, AdaLoRA achieves an average score of 4.96, and its quantized variant, Q-AdaLoRA, is rated at 4.60 (−0.36). These variances are minimal, especially considering the scale of ratings ranging from one to ten.

Qualitative evaluation. Table IV shows the average grade for answers of all fine-tuned models to our five selected questions (see Section III-A3). We observe that, on average, LoRA performs the best, with a strong performance gap to AdaLoRA that in turn outperforms LoKr. Interestingly, the human evaluation of the answers does indicate a quite steep drop-off in quality between the quantized and unquantized versions of LoRA and AdaLoRA. This is a further indication that model quantization does not incur just a negligible loss in performance as most automated evaluations suggest.

D. Colab Notebooks to “Run Your LLM at Home”

We provide easy-to-use Colab notebooks for all presented methods that work on a T4 GPU (see https://github.com/tuggeluk/LMM_at_home). Using these notebooks, all quan-

TABLE IV
AVERAGE (ACROSS 8 HUMAN RATERS) USEFULNESS EVALUATION (1=USELESS, 5=PERFECT) OF ANSWERS FROM FINE-TUNED LLAMA V2S.

	Q1	Q2	Q3	Q4	Q5	Average
LoRA	2.5	2.25	0.75	3.75	3.25	2.5
Q-LoRA	2	0.75	2.25	3.25	1.75	2
AdaLoRA	1.5	1.25	2.5	3	2.25	2.1
Q-AdaLoRA	1.25	1.75	1.25	1.5	2.25	1.6
LoKr	2.25	1.5	1	0.5	3	1.65

tized inference methods as well as Q-LoRA and Q-AdaLoRA fine-tuning can be easily deployed on a Colab free-tier instance by anyone without any coding.

IV. CONCLUSIONS

We have surveyed the state of the art in methods for efficient fine-tuning and inference with LLMs, and have shown that the current ecosystem of open-source LLMs with accompanying resource-sparse technologies allows for efficient and effective LLM inference and fine-tuning with limited computational resources. As expected, quantization led to a massive reduction in GPU memory requirements. For inference, this has also carried over to a significant reduction in power consumption which might be an important factor for large operations. Most quantization methods claim to cause a negligible loss in performance. However, we have been able to show, to the best of our knowledge for the first time, that quantization has caused a noticeable degradation in performance for both inference and fine-tuning on LlamaV2 models. Important to note here is that the automated evaluation was not able to accurately capture this shift in performance, rather, it was only found thanks to the qualitative evaluation using human input. This further enforces an emerging sentiment in the community that the current evaluation methods for generative LLMs are not sufficient [55]. We have further shown that low-rank-adaptation-based fine-tuning effectively trains LLMs. AdaLoRA and LoKr were not able to outperform the default of LoRA (again in the context of LlamaV2 models), which calls into question whether these further developments are useful, especially the computationally heavy LoKr. We furthermore observed that our fine-tuned models with short training perform worse than LlamaV2-Chat, which shows that fine-tuning still is a lengthy process and highly-tuned open source models are usually preferable over home-brewed variants where available.

ACKNOWLEDGMENTS

This work was partially funded by a DIZH Fellowship of LT. We thank Thomas Graf, Pascal Schneider, Simon Spalinger, and Daniel Tuggener for contributing as LLM raters.

REFERENCES

- [1] W. X. Zhao *et al.*, “A Survey of Large Language Models,” Nov. 2023. arXiv:2303.18223.
- [2] J. Segessenmann, T. Stadelmann, A. Davison, and O. Dürr, “Assessing deep learning: a work program for the humanities in the age of artificial intelligence,” *AI and Ethics*, pp. 1–32, 2023.
- [3] T. Stadelmann, T. Klamt, and P. H. Merkt, “Data centrism and the core of data science as a scientific discipline,” *Archives of Data Science, Series A*, vol. 8, no. 2, 2022.
- [4] A. Vaswani *et al.*, “Attention is All you Need,” in *Proc. of NIPS’17*, vol. 30.
- [5] I. Schlag, K. Irie, and J. Schmidhuber, “Linear Transformers Are Secretly Fast Weight Programmers,” in *Proc. of ICML’21*, vol. 139, pp. 9355–9366.
- [6] J. Kaplan *et al.*, “Scaling Laws for Neural Language Models,” Jan. 2020. arXiv:2001.08361.
- [7] H. Touvron *et al.*, “Llama 2: Open Foundation and Fine-Tuned Chat Models,” July 2023. arXiv:2307.09288.
- [8] G. Menghani, “Efficient deep learning: A survey on making deep learning models smaller, faster, and better,” *ACM Computing Surveys*, vol. 55, no. 12, pp. 1–37, 2023.
- [9] L. Tuggener, M. Amirian, F. Benites, P. von Däniken, P. Gupta, F.-P. Schilling, and T. Stadelmann, “Design patterns for resource-constrained automated deep-learning methods,” *AI*, vol. 1, no. 4, pp. 510–538, 2020.
- [10] T. Dettmers, M. Lewis, Y. Belkada, and L. Zettlemoyer, “LLM.int8(): 8-bit Matrix Multiplication for Transformers at Scale,” in *NeurIPS’22*, vol. 35, pp. 30318–30332.
- [11] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, “QLoRA: Efficient Finetuning of Quantized LLMs,” vol. 36, pp. 10088–10115.
- [12] E. Frantar, S. Ashkboos, T. Hoefler, and D. Alistarh, “GPTQ: Accurate Post-Training Quantization for Generative Pre-trained Transformers,” *Proc. of ICLR’23*.
- [13] J. Lin *et al.*, “AWQ: Activation-aware Weight Quantization for LLM Compression and Acceleration,” *Proc. of MLSys’24*.
- [14] T. Dettmers, M. Lewis, S. Shleifer, and L. Zettlemoyer, “8-bit Optimizers via Block-wise Quantization,” in *Proc. of ICLR’22*.
- [15] E. J. Hu *et al.*, “LoRA: Low-Rank Adaptation of Large Language Models,” in *Proc. of ICLR’22*.
- [16] Q. Zhang *et al.*, “AdaLoRA: Adaptive Budget Allocation for Parameter-Efficient Fine-Tuning,” *Proc. of ICLR’23*.
- [17] A. Edalat *et al.*, “KronA: Parameter Efficient Tuning with Kronecker Adapter,” Dec. 2022. arXiv:2212.10650.
- [18] Y. LeCun, J. Denker, and S. Solla, “Optimal Brain Damage,” *Proc. of NIPS’89*, vol. 2, 1989.
- [19] B. Hassibi, D. G. Stork, and G. J. Wolff, “Optimal brain surgeon and general network pruning,” in *Proc. of the IEEE ICNN’93*, pp. 293–299.
- [20] S. Han, H. Mao, and W. J. Dally, “Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding,” *Proc. of ICLR’15*.
- [21] M. Sun, Z. Liu, A. Bair, and J. Z. Kolter, “A Simple and Effective Pruning Approach for Large Language Models,” Oct. 2023. arXiv:2306.11695.
- [22] E. Frantar and D. Alistarh, “SparseGPT: Massive Language Models Can Be Accurately Pruned in One-Shot,” in *Proc. of ICML’23*, pp. 10323–10337.
- [23] J. Rasley, S. Rajbhandari, O. Ruwase, and Y. He, “DeepSpeed: System Optimizations Enable Training Deep Learning Models with Over 100 Billion Parameters,” in *Proc. of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 3505–3506, 2020.
- [24] S. Rajbhandari, J. Rasley, O. Ruwase, and Y. He, “ZeRO: Memory Optimizations Toward Training Trillion Parameter Models,” in *Proc. of SC’20*, pp. 1–16.
- [25] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” in *Proc. of ICLR’15*.
- [26] I. Loshchilov and F. Hutter, “Decoupled Weight Decay Regularization,” in *Proc. of ICLR’19*.
- [27] I. Loshchilov and F. Hutter, “SGDR: Stochastic Gradient Descent with Warm Restarts,” in *Proc. of ICLR’17*.
- [28] T. Chen, B. Xu, C. Zhang, and C. Guestrin, “Training Deep Nets with Sublinear Memory Cost,” Apr. 2016. arXiv:1604.06174.
- [29] A. Gruslys *et al.*, “Memory-Efficient Backpropagation through Time,” in *Proc. of NIPS’16*, pp. 4132–4140.
- [30] S. Gugger, L. Debut, T. Wolf, P. Schmid, Z. Mueller, S. Mangrulkar, M. Sun, and B. Bossan, “Accelerate documentaion,” 2022.
- [31] G.-I. Yu *et al.*, “Orca: A Distributed Serving System for Transformer-Based Generative Models,” in *USENIX Symposium on Operating Systems Design and Implementation*, pp. 521–538, July 2022.
- [32] F. Jelinek, R. L. Mercer, L. R. Bahl, and J. K. Baker, “Perplexity—a measure of the difficulty of speech recognition tasks,” *The Journal of the Acoustical Society of America*, vol. 62, pp. S63–S63, Dec. 1977.
- [33] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “Bleu: a Method for Automatic Evaluation of Machine Translation,” in *Proc. of ACL’02*, pp. 311–318.
- [34] S. P. Singh *et al.*, “Machine translation using deep learning: An overview,” in *International Conference on Computer, Communications and Electronics*, Compelix, pp. 162–167, July 2017.
- [35] S. Minaee *et al.*, “Deep Learning-based Text Classification: A Comprehensive Review,” *ACM Computing Surveys*, vol. 54, pp. 1–40, Apr. 2022.
- [36] R. Anil *et al.*, “PaLM 2 Technical Report,” Sept. 2023. arXiv:2305.10403.
- [37] S. Bubeck *et al.*, “Sparks of Artificial General Intelligence: Early experiments with GPT-4,” Apr. 2023. arXiv:2303.12712.
- [38] A. Srivastava *et al.*, “Beyond the Imitation Game: Quantifying and extrapolating the capabilities of language models,” June 2023. arXiv:2206.04615.
- [39] P. Clark *et al.*, “Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge,” Mar. 2018. arXiv:1803.05457.
- [40] R. Zellers *et al.*, “HellaSwag: Can a Machine Really Finish Your Sentence?,” in *Proc. of ACL’19*, pp. 4791–4800, 2019.
- [41] D. Hendrycks *et al.*, “Measuring Massive Multitask Language Understanding,” *Proc. of ICLR’21*.
- [42] K. Sakaguchi, R. L. Bras, C. Bhagavatula, and Y. Choi, “WinoGrande: An Adversarial Winograd Schema Challenge at Scale,” *Communications of the ACM*, vol. 64, pp. 99–106, Aug. 2021.
- [43] M. Chen *et al.*, “Evaluating Large Language Models Trained on Code,” July 2021. arXiv:2107.03374.
- [44] K. Cobbe *et al.*, “Training Verifiers to Solve Math Word Problems,” Nov. 2021. arXiv:2110.14168.
- [45] W. Zhong *et al.*, “AGIEval: A Human-Centric Benchmark for Evaluating Foundation Models,” Sept. 2023. arXiv:2304.06364.
- [46] J. Wei *et al.*, “Finetuned Language Models Are Zero-Shot Learners,” in *Proc. of ICLR’22*.
- [47] S. Mishra, D. Khashabi, C. Baral, and H. Hajishirzi, “Cross-Task Generalization via Natural Language Crowdsourcing Instructions,” in *Proc. of ACL’22*, pp. 3470–3487.
- [48] Y. Wang *et al.*, “Super-NaturalInstructions: Generalization via Declarative Instructions on 1600+ NLP Tasks,” in *Proc. of EMNLP’22*, pp. 5085–5109, Dec.
- [49] S. Longpre *et al.*, “The Flan Collection: Designing Data and Methods for Effective Instruction Tuning,” vol. 202 of *Proc. of ICML’23*, pp. 22631–22648.
- [50] Y. Wang *et al.*, “Self-Instruct: Aligning Language Models with Self-Generated Instructions,” in *Proc. of ACL’23*, pp. 13484–13508.
- [51] S. Reddy, D. Chen, and C. D. Manning, “CoQA: A Conversational Question Answering Challenge,” *Trans. of ACL*, vol. 7, pp. 249–266, 2019.
- [52] J. Feng *et al.*, “MMDialoG: A Large-scale Multi-turn Dialogue Dataset Towards Multi-modal Open-domain Conversation,” Dec. 2022. arXiv:2211.05719.
- [53] A. Köpf *et al.*, “OpenAssistant Conversations – Democratizing Large Language Model Alignment,” Oct. 2023. arXiv:2304.07327.
- [54] L. Zheng *et al.*, “Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena,” Oct. 2023. arXiv:2306.05685.
- [55] J. Deriu, D. Tuggener, P. Von Däniken, and M. Cieliebak, “Probing the Robustness of Trained Metrics for Conversational Dialogue Systems,” in *Proc. of ACL’22*, pp. 750–761.

Originally published in IEEE Access, May 2024, DOI 10.1109/ACCESS.2024.3404834.



Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2023.1120000

MathNet: A Data-Centric Approach for Printed Mathematical Expression Recognition

FELIX M. SCHMITT-KOOPMANN^{1, 2}, ELAINE M. HUANG², HANS-PETER HUTTER¹ (Member, IEEE), THILO STADELMANN^{3, 4} (Senior Member, IEEE), ALIREZA DARVISHY¹

¹Institute of Computer Science, ZHAW, 8401 Winterthur, Switzerland²People and Computing Laboratory, University of Zurich, 8050 Zurich, Switzerland³Centre for Artificial Intelligence, ZHAW, 8400 Winterthur, Switzerland⁴European Centre for Living Technology (ECLT), 30123 Venice, Italy

Corresponding author: Felix M. Schmitt-Koopmann (e-mail: scm@zhaw.ch).

This work was supported by the Bridge Discovery program of the Swiss National Science Foundation under grant number 194677.

ABSTRACT Printed mathematical expression recognition (MER) models are usually trained and tested using LaTeX-generated mathematical expressions (MEs) as input and the LaTeX source code as ground truth. As the same ME can be generated by various different LaTeX source codes, this leads to unwanted variations in the ground truth data that bias test performance results and hinder efficient learning. In addition, the use of only one font to generate the MEs heavily limits the generalization of the reported results to realistic scenarios. We propose a data-centric approach to overcome this problem, and present convincing experimental results: Our main contribution is an enhanced LaTeX normalization to map any LaTeX ME to a canonical form. Based on this process, we developed an improved version of the benchmark dataset *im2latex-100k*, featuring 30 fonts instead of one. Second, we introduce the real-world dataset *realFormula*, with MEs extracted from papers. Third, we developed a MER model, *MathNet*, based on a convolutional vision transformer, with superior results on all four test sets (*im2latex-100k*, *im2latexv2*, *realFormula*, and *InfyMDB-1*), outperforming the previous state of the art by up to 88.3%.

INDEX TERMS Data-Centric AI, Deep Learning, Labeling, Document Analysis, Mathematical Expression Recognition, Pattern Recognition

I. INTRODUCTION

Recognizing mathematical expressions (MEs) in images and converting them into a machine-understandable format is known as mathematical expression recognition (MER). Creating a dependable MER would unlock possibilities for producing innovative tools, such as the ability to digitize, search, extract, and enhance the accessibility of mathematical equations in documents [1].

However, despite recent progress in the field of MER, it remains a challenge for two main reasons. Firstly, MEs contain many symbols, i.e., multiple alphabets, numerals, operators, and parentheses. Secondly, structural information (for example, nested superscripts and subscripts) is crucial for correctly recognizing MEs [2], [3].

In addition, we have identified a third challenging aspect that needs to be addressed. The machine-understandable format used in many MER models can cause unwanted variation. For instance, LaTeX is a popular format used by many MER models [4]. However, LaTeX allows authors to write the same ME with different LaTeX code as shown in Figure 1. Accordingly,

many LaTeX commands are redundant or can be neglected without altering the canonical form or even without changing the visual appearance of an ME. For example, we observed that of the 500 different tokens in the printed MER benchmark dataset *im2latex-100k* [5], 174 tokens or 34.8% of the vocabulary is redundant or does not influence the canonical form of the ME. This leads to detrimental variability in the training data and therefore to inefficient learning, excessive training data needs, and, finally, suboptimal recognition performance due to unresolved ambiguity in the model [6], [7]. Finally, the use of a single font for MEs in *im2latex-100k* heavily limits the generalization of performance results reported on this data set to realistic scenarios.

Reducing this variability is not only to reduce unwanted biases in test scores but is expected to have a high impact on learning quality of respective models and, hence, their performance. Recently, methods have emerged under the term data-centric AI that systematically engineer the data in order to improve overall system quality [8], [9]. The methodology is characterized by making the data a first-class citizen in the

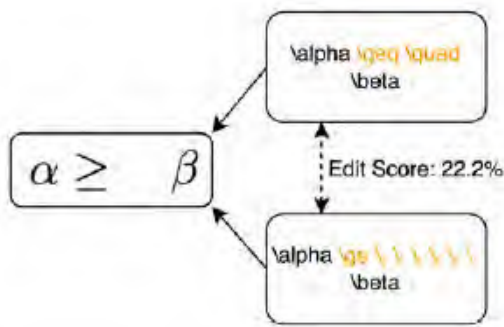


FIGURE 1. An example of an ME image that can be produced with more than one LaTeX code. While the two presented LaTeX codes are quite different (22.2% Edit score), they create the same image.

development process of any machine-learning-based system, thus shifting the focus away from merely manipulating the model architecture [10]. In this paper, we adopt a data-centric approach by proposing a systematic process to map an ME to a canonical LaTeX representation. Since we develop our methods to make MEs in PDFs accessible, we focus in this work on printed MER, but our approaches are also applicable to handwritten MEs and we expect similar benefits for complex handwritten MEs. We present the following major contributions: 1) A LaTeX normalization process that maps LaTeX MEs to a canonical form. 2) *Im2latexv2*, an upgraded version of *im2latex-100k* with multiple fonts and a canonical ground truth (GT). 3) *realFormula*, a real-world test set for MER. 4) Our MER model *MathNet* which outperforms the previous state of the art on all four test sets by up to 88.3%. The remainder of this paper is structured as follows: we present related work in Section II. We discuss the issues with using LaTeX for MER in Section III. We introduce the datasets we have developed, the metrics used, and our printed-MER model in Section IV. We will then present the results of our experiments in Section V and discuss them in Section VI. Finally, we offer concluding remarks in Section VII.

II. RELATED WORK

MER has been a research task for over 50 years [11], and it still remains open. Although the focus of the MER research field has shifted to the recognition of online and offline handwritten MEs in the last years, research on printed MEs is still important to make it applicable in practice. The two fields of MER research overlap, but there are also two major differences. First, the offline handwritten MER has the extra challenge of touching symbols, which makes it harder to separate them [3], [4]. Second, the characteristics of the benchmark datasets are different. Handwritten (offline) MER uses the CROHME datasets [12] as the benchmark, with a vocabulary of 142 tokens and, on average, 18 tokens per ME. On the other hand, the printed MER benchmark dataset *im2latex-100k* [5] has a much larger vocabulary of 500 tokens, which is 3.5 times greater than the CROHME dataset. Additionally, on

average, each ME in the *im2latex-100k* dataset has 2.8 times as much tokens.

However, both MER systems comprise three stages: symbol segmentation, symbol recognition, and 2D structure analysis [4]. Classic approaches, as the Infy system [13], [14] solve these stages separately, whereas end-to-end approaches address them all at once. With recent progress in deep learning, end-to-end approaches with an encoder-decoder structure have become prevalent [15]. These systems directly map input images to a semantic text representation, e.g., LaTeX. In general, the encoder is based on convolutional layers to calculate features of the image. The decoder generally uses LSTMs [16], GRUs [17], or Transformers [18], which translate the feature inputs step-by-step into a token sequence [4]. *WYGIWYS*, introduced by Deng et al. [15], is one of the first end-to-end MER systems. It calculates its features using a convolutional network stacked with an RNN row encoder. The token sequence is predicted by an RNN decoder with visual attention stacked with a classifier layer. Because of the end-to-end approach, large datasets are required for training. Therefore, the authors introduced *im2latex-100k* [5], which is still the classic benchmark dataset in printed MER.

Cho et al. [19] found that the performance of the encoder-decoder network for text generation declines as the length of the sentence increases. This is particularly relevant for ME sequences, which are usually longer than sentences used in image captioning. As a result, many MER models focus on enhancing the long-distance dependence of the decoder. Various approaches have been developed to overcome this issue.

Bian et al. [20] developed a bi-directional mutual learning network based on attention aggregation. The network uses two encoders, one that processes the input left-to-right and another that processes it right-to-left. They demonstrated that this structure helps alleviate the issue of long-range dependencies in RNNs. Li et al. [21] introduced a method for counting symbols in handwritten MER. Their weakly supervised multi-scale counting module can be combined with most encoder-decoder frameworks, and it improves the model's robustness when the ME is complex and/or long. However, it does not solve the problem with variations in writing styles. Yan et al. [22] developed *ConvMath*, a printed MER system based entirely on convolutions. They introduced a convolutional decoder to better detect the 2D relation of MEs. Markazemy et al. [23] introduced a novel reinforcement learning module to process the decoder output and refine it. Apart from focusing on the decoder, various elements of MER have been researched. Wang et al. [24] aimed to enhance the encoder by incorporating DenseNet into printed MER. Li et al. [25] introduced scale augmentation and drop attention to handwritten MER to improve the model performance for various ME scales. Peng et al. [26] introduced Graph Neural Network in printed MER. However, representing an ME as a graph became popular in handwritten MER [27], but not in printed MER [4]. Singh [28] investigated the visual attention in printed MER and developed two new datasets based on

im2latex-100k. Furthermore, there have been advancements in the development of end-to-end systems for scientific documents that can recognize not only MEs but also text and tables. However, the current leading end-to-end system from Blecher et al. [29] has an Edit distance of only 87.2%, which is lower than the best current MER systems.

However, to the best of our knowledge, the influence of undesired variations in the GT has not yet been investigated in handwritten nor printed MER.

III. DETRIMENTAL LATEX VARIATIONS

MEs have a two-dimensional structure which is different from the one-dimensional structure of natural language text. Therefore, a markup language, e.g. LaTeX, is needed to convert MEs into a natural language description. LaTeX is widely used in the scientific community for writing documents. Hence, many MEs in LaTeX exist, making it appealing for printed MER. The widely used benchmark dataset *im2latex-100k* also uses LaTeX to create the MEs for the training and test datasets. However, we discovered two problematic issues with this dataset:

- 1) Our analysis of the dataset revealed that the whole *im2latex-100k* dataset was created with a single font. This, on the one hand, drastically limitates the generalisation capability of the performance results reported on this dataset to realistic scenarios where MEs are printed in various different font styles, normally different than the one used for training the systems. This effect was revealed in preliminary experiments when we observed a significant decrease in the performance of all tested systems by changing the font of the test set. This effect is also apparent when we compare the performance results of the baseline models of *im2latex-100k* and *im2latexv2* (refer to Tables 4 and 5). To address this limitation, all MEs of the *im2latex-100k* dataset were generated in 16 different fonts.
- 2) We further discovered another detrimental effect in the GT of *im2latex-100k*: As the GT of the MEs in *im2latex-100k* was taken from real papers written by different authors, there was a large variation of the GT for semantically identical MEs, as illustrated in Figure 1. These variations have nothing to do with improved generalization capabilities to be learned or shown. To the contrary: First, it reduces the validity of performance result comparisons of the different systems if this occurs in the test dataset. Second, it is detrimental for the learning of MER systems, if it occurs in the training dataset (by teaching the model that the same input has ambiguous output, leading to reduced learning [30]). In order to minimize these meaningless variations in the GT of *im2latex-100k*, we adopted a data-centric approach to develop a new LaTeX normalization procedure. The data-centric approach involves three steps. First, the model is trained using the existing training data. Second, the performance of the trained model is evaluated to identify any error patterns. Third, these er-

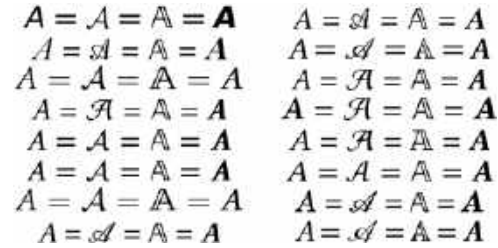


FIGURE 2. The ME: $A = \mathcal{A} = \mathbb{A} = \mathbf{A}$ generated with the 16 fonts, which can render all three basic mathematical fonts.

ror patterns are utilized to improve the training dataset (in our case by adjusting the LaTeX normalization).

These steps are repeated until no more error patterns can be detected. During this iterative process, we have identified six problematic aspects in the GT of the *im2latex-100k* dataset: mathematical fonts, white spaces, curly brackets, sub- and superscript order, tokens, and arrays. These problematic aspects together with our proposed solutions are described in Sections III-A - III-F. We designed our normalization process to address these issues and reduce undesired variations. The normalization algorithm is publicly accessible via GitHub [31].

A. MATHEMATICAL FONTS

Using different mathematical fonts, such as bold or double bold, to indicate vectors or spaces can be challenging for MER. Recognizing these mathematical fonts is simple if only one font is used for all MEs, but it becomes challenging with multiple fonts, as shown in Figure 2. Additionally, it can be challenging to create a dataset with mathematical fonts, as not all mathematical font commands work with every font, i.e., only 16 out of 59 fonts respond to the three basic mathematical font commands ($\mathcal{A}$, $\mathbb{A}$, and $\mathbf{A}$) for all symbols. As a result, the collected ME can contain a mathematical font command that does not influence the compiled image of the ME. To avoid this, we decided to remove all mathematical font commands, which is a simplification of the task but reduces the number of labeling errors in the GT.

B. WHITE SPACES

In LaTeX, authors can adjust the white space between two symbols using various commands (e.g., $\quad$). However, these commands are primarily defined relative to the font size, making it essential for the model to accurately detect the font size, which is influenced by the font. Additionally, multiple combinations of white space commands exist for each relative white space length. This makes it impossible for the model to predict the white space commands when multiple fonts are utilized, and the white space commands do not follow a clear pattern. Since the white space does not impact the canonical form of an ME, we decided to remove all white space commands from the GT.

C. CURLY BRACKETS

In LaTeX, curly brackets are used to define the scope of LaTeX commands. As a result, 33% of all tokens in *im2latex-100k* are curly brackets. However, the issue with curly brackets is that they are often optional and can be added without changing the visual appearance of a mathematical expression (e.g., a_3 , $a_{\{3\}}$, and $\{a_{\{3\}}\}$ are visually identical). Therefore, we introduced a precise definition of which curly brackets are required and which are not. This reduces ambiguity and the number of curly brackets in the GT.

D. SUB- AND SUPERScript ORDER

Symbols can have sub- and superscripts but the order in LaTeX code is irrelevant for the visual appearance of the ME. If multiple sub- and superscripts (e.g., $a^{\{b\}}_{\{c\}}^{\{d\}}$) exist, we decided to combine these to one subscript and one superscript (e.g., $a_{\{c\}}^{\{bd\}}$) to reduce ambiguity and the number of tokens. Although this normalization steps may result in errors in certain circumstances, it typically minimizes undesired variations of the GT.

E. TOKENS

We identified three issues on the token level. First, many expressions in LaTeX exist that produce the same visual symbol (e.g., $\geq$ and $\geqslant$). Hence, we identified all redundant LaTeX expressions in *im2latex-100k* and replaced them by the canonical form. Second, some tokens in the ME imply that the ME not only contains mathematical elements (e.g., $\text{\cite{}}$, $\text{\label{}}$) or is more a graphic element than an ME (e.g., $\text{\fbox{}}$). Hence, we decided to delete all MEs with such tokens. Third, the tokenizer introduced by Deng et al. [15] sometimes combines two LaTeX commands in one token, e.g., the token $\text{\right\}}$ actually contains two tokens $\text{\right}$ and $\{$. This can increase the vocabulary and introduce undesired variation. To avoid this, we split up these tokens, so each token represents only one LaTeX command.

F. ARRAYS

The array structure has the purpose to arrange elements in a grid, e.g., a matrix. However, many authors use this feature to align MEs instead (e.g., $\begin{array}{cc} a=b, & c=d \end{array}$). Additionally, the column alignment indicators (l, c, r) do not affect the semantics of the array. Moreover, not all arrays are well-defined and may contain empty columns, rows, or cells. Hence, we removed array structures used only for formatting, and reduced GT variation in the array structures by replacing all column indicators with c. We also removed sparse arrays (empty entries or number of columns doesn't match number of column alignment indicators).

IV. APPLYING LATEX NORMALIZATION FOR PRINTED MER

To evaluate our LaTeX normalization, we applied it for printed MER. Therefore, we developed an enhanced version of *im2latex-100k* described in Section IV-A, a real-world

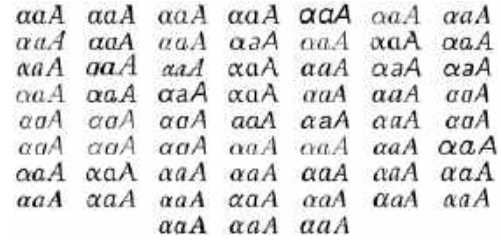


FIGURE 3. Overview of all 59 fonts in the *im2latexitv2* dataset.

test set described in Section IV-B, and a new printed MER model described in Section IV-C. Lastly, Section IV-D gives an introduction to printed MER metrics.

A. IM2LATEXV2

This dataset is an evolution of *im2latex-100k* and contains three major modifications over existing printed MER datasets. First, we used the normalization procedure described in Section III with minor modifications for rendering. To create controlled visual diversity, we left the column alignment indicators of arrays unchanged and did not remove the $\text{\right}$ and $\text{\left}$ tokens for rendering the MEs. Using the normalized MEs we can ensure that the GT and image coincide. In comparison, Deng et al. [15] used the original ME descriptions for *im2latex-100k*. Hence, the GT for the same image may vary.

Second, in contrast to *im2latex-100k*, *im2latex-90k*, and *im2latex-140k*, we rendered each ME with 30 different fonts for the training dataset and 59 for the validation and test set. The incorporation of multiple fonts makes the dataset more realistic. Furthermore, 29 fonts only appear in the validation and test set to assess a model's generalization capability. The font variation introduced this way is illustrated in Figure 3.

Third, we used 600 DPI (font size 12pt) to render the images, because down-sampling works well compared to up-sampling. In contrast, Deng et al. [15] suggested 100 DPI for the MER task. Singh [28] used 200 DPI and in handwritten MER different resolutions exist. However, the scanned images of handwritten MER correspond to resolutions of 300 to 600 DPI in printed MER. We will demonstrate the influence of the resolution on the model performance in Section V-A. The resulting *im2latexitv2* dataset contains fewer MEs than the original *im2latex-100k* due to our rendering pipeline, which includes four check criteria (see Algorithm 1 and Table 1). 19 MEs in the training set and 30 MEs in the test set had to be dropped because the image was blank. Additionally, we found 1 empty ME in the train set and 42 empty MEs in the test set. We manually corrected the empty ME in the train set and 37 MEs in the test set. We removed the other 5 empty MEs from the test set because the image depicted a drawing rather than a valid ME. Besides, our normalization step dropped 882 MEs in the training set, 116 in the validation set, and 179 MEs in the test set. The rendering step removed 129 MEs in the

TABLE 1. Overview of the reasons, why we deleted different MEs.

Category	Alg. 1	train	validation	test
im2latex-100k		75'275	8'370	10'355
white image	2	19	0	30
empty ME (corrected)	5	1 (1)	0	42 (37)
normalization step	10	882	116	179
rendering errors	15	129	11	23
im2latexv2		74'245	8'243	10'118

training set, 11 MEs in the validation set, and 23 MEs in the test set, which could not be rendered for all fonts. As a result, the training set was reduced by 1'023 MEs, the validation set by 127 MEs, and the test set by 237 MEs compared to the original *im2latex-100k*. The new normalized dataset *im2latexv2* finally contains approximately 92'600 MEs (ref. Table 1). It is publicly available on GitHub [31].

Algorithm 1 Im2latexV2 Rendering Pipeline

Require: $[F_1, I_1] \dots [F_N, I_N]$ in *im2latex-100k*

```

1: for  $k \leftarrow 1$  to  $N$  do
2:   if  $I_k$  is null then
3:     continue
4:   end if
5:   if  $F_k$  is null then
6:     check manually
7:     continue
8:   end if
9:    $F_k \leftarrow \text{LaTeXNormalization}(F_k)$ 
10:  if  $F_k$  is null then
11:    continue
12:  end if
13:  for  $j, r$  in renderingSetups do
14:     $I_{k,j}, e_{k,j} \leftarrow \text{renderImages}(F_k, \text{renderingSetup})$ 
15:    if  $e_{k,j}$  is not null then
16:      break
17:    end if
18:  end for
19: end for

```

B. REALFORMULA

By using the Mathematical Formula Detection model from Schmitt-Koopmann et al. [32], we collected over 250k ME from randomly selected arXiv papers with 600 DPI and selected 200 MEs at random for manual annotation. As shown in Table 2 we deleted 69 MEs where the image was larger than 768x2400 pixels. Nine other MEs were deleted because the image did not show the complete ME and 1 ME showed a sparse matrix. Hence, we manually annotated 121 MEs. Of these 121 MEs, 110 were single-line MEs and 11 were multi-line MEs. Five single-line MEs contained an array, and 43 MEs contained style types (`\boldsymbol`, `\mathbb`, `\mathcal`). *realFormula* is publicly available at GitHub [31].

TABLE 2. Overview of 200 randomly selected MEs. It shows various issues that arose, requiring some MEs to be excluded from the *realFormula* set.

Category	Number of images
too large	69
cropping error	9
sparse matrix	1
removed	79
correct	121

C. MATHNET

For our experiments we decided to use an encoder-decoder approach similar to the state of the art MER models.

In order to accurately process ME images, it is crucial for the encoder to extract informative features. This requires the encoder to be able to focus on small details while also considering the overall structure of the ME, such as a fraction. To handle both short-term and long-term relationships, Deng et al. [15] developed the Coarse-to-Fine Attention mechanism. However, recent advancements in image recognition have shown that vision transformers (ViTs) [33] are well-suited for this task. A further development of ViTs are convolutional vision transformers (CvTs) [34]. CvTs combine convolutions with transformers, resulting in superior performance and efficiency with a smaller model. Hence, we decided to use a CvT instead of a usual CNN encoder.

The decoder is responsible for converting the features of the encoder into the chosen markup language, i.e. LaTeX. Unlike most other MER systems, *MathNet* uses a regular decoder transformer instead of LSTMs. Vaswani et al. [18] showed that transformers are better suited for handling long sequences, as we have in printed MER. Furthermore, *im2latexv2* is much larger than *im2latex-100k*, which should benefit the training of transformers. Our decoder transformer has 8 heads and a depth of 4. On top of this, we added a classifier layer with a log softmax. An overview of our *MathNet* model with the layer sizes is shown in Fig. 4.

We used a cross-entropy loss between the GT sequence and the predicted sequence. To optimize our model, we used the Adam optimizer [35] with an initial learning rate of 0.000075 and a batch size of 36. Our model was trained on a single Nvidia Tesla V100-SXM2-32GB GPU.

In order to prevent the model from learning useless patterns in images, we applied different augmentation techniques. We have identified four patterns that the model should explicitly not learn. Firstly, to avoid confusion by white spaces (such as `\quad` and `\,`), we randomly introduced white pixel columns to the image. Secondly, we used blurring masks and changed the image resolution randomly. Thirdly, we resized the image to prevent the model from focusing on a specific text size. Lastly, we added a white border of variable size to the image to facilitate batch-wise processing and ensure that all images have the same size.

D. METRICS

Printed MER primarily uses three metrics (Edit distance, Bleu-Score, and Exact Match) to evaluate model perfor-

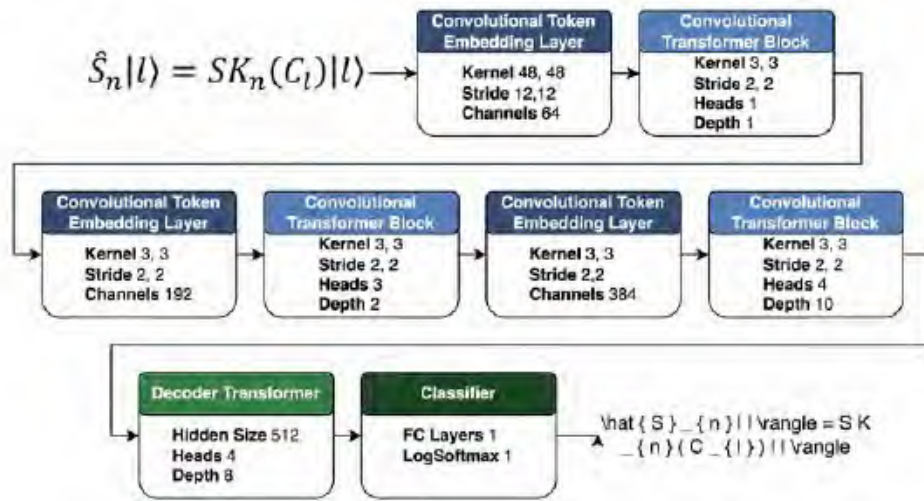


FIGURE 4. Overview of our MER model, called *MathNet*. The CvT consists of 3 layers, which are a combination of an embedding layer and a transformer block. The encoded image is decoded with a decoder transformer and a classifier layer.

mance. An edit distance counts the operations needed to transform one sequence into another sequence. Depending on the operations allowed, different edit distances exist. The most popular edit distance for printed MER and the one we use is the Levenshtein edit distance (*lev*). It contains three operations: 1) insert a new token, 2) delete a token, and 3) replace a token. The Edit score, as used by Deng et al., is the edit distance normalized by the max sequence length of the GT and predicted sequence (PRE) as shown in Eq. 1. For MER, an Edit score of 100% is a perfect prediction.

$$\text{EditScore} = \left(1 - \frac{\text{lev}(\text{GT}, \text{PRE})}{\max(\text{len}(\text{GT}), \text{len}(\text{PRE}))}\right) \cdot 100\% \quad (1)$$

The Bleu score compares subsequences of two sequences. A predefined number, usually 4 in MER, determines the maximum subsequence length. To determine the Bleu score you create n-grams for the sequences and then calculate the precision between the n-grams of sequences. The Bleu score is the average precision with a brevity penalty to discourage overly short predictions. However, the Bleu score is designed for longer sequences, and errors at the sequence borders count less than errors in the middle. This behavior can significantly impact the score of MEs. Exact Match measures the amount of fully correct MEs. It makes no distinction between partially correct ME and completely incorrect ME.

With respect to PDF accessibility, we regard the Edit score as the most relevant metric because it indicates the amount of work that must be manually done to correct all errors in a recognized ME. The Bleu-4 score shows unwanted behavior for short MEs and focuses more on patterns than on the correct order of the tokens. The interpretation of exact match is very limited through the binary output.

However, as discussed in Section III, semantically identical MEs can be produced with different LaTeX code sequences.

Hence, the metric results are largely influenced by the used tokenizer and normalization. For example, if the x in Eq. 2 should be a 5, the Edit score would rise from 96.3% (1 of 27) to 97.8% (1 of 45) if curly brackets had been added in the GT around each entry in the array, which would mean a 40.5% reduction of the Edit error rate (1 - Edit score). The Bleu-4 score shows similar behavior.

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & x & 6 \\ 7 & 8 & 9 \end{pmatrix} \quad (2)$$

V. EXPERIMENTS WITH PRINTED MER

This section presents the results of four experiments. The first experiments (see Section V-A) demonstrate the influence of the image resolution on the model performance. Experiments two to five are comparison experiments with printed MER models. To ensure a fair comparison, we used the provided pre-trained models (*WYGIWYS*, *i2l-strips*, *i2l-nopool*) and normalized the predictions with our normalization process. We used the following four datasets to compare the models: 1) The benchmark dataset *im2latex-100k*. 2) Our enhanced version *im2latev2* which includes multiple fonts. 3) Our developed real-world dataset *realFormula* with MEs images extracted from papers, to demonstrate how well the systems perform in a real-world environment. 4) *InfyMDB-1* contains MEs images, which were scanned with 600 DPI. This dataset is also intended to evaluate real-world performance, specifically the impact of scanning noise.

It is important to note that *i2l-strips* and *i2l-nopool* used a modified dataset built upon *im2latex-100k* which had a different split between training, validation, and testing. Hence, MEs of the test sets of *im2latex-100k* and *im2latev2* could be in the training set of *i2l-strips* and *i2l-nopool*.

TABLE 3. Influence of the training image resolution on the *im2latexv2* test set. We trained our model with 100, 200, 300, and 600 DPI.

DPI	Edit [%]	Bleu-4 [%]	EM [%]
100	78.2	66.0	6.0
200	93.5	93.2	56.9
300	96.9	96.9	75.0
600	98.0	98.2	84.9

A. OPTIMAL IMAGE RESOLUTION

When the resolution is low, the image has fewer details and the model has to focus on the general structure. On the other hand, high-resolution images provide more detail which can enable the model to differentiate better between symbols. However, there is no clear definition what the optimal image resolution for MER is (with the standard font size of 12pt). According to Deng et al. [15], 100 DPI images are recommended, while Singh [28] used images with 200 DPI. In contrast, handwritten MER mainly uses image sizes that correspond to resolutions between 300 and 600 DPI [12]. We trained our model on various image resolutions to demonstrate the impact of this resolution, as shown in Table 3. We used 100, 200, 300, and 600 DPI image resolutions. The images of the test set were scaled accordingly. Our results reveal a significant improvement between 100 and 200 DPI. Moreover, the model's performance still improves with even higher resolutions. However, we did not test resolutions higher than 600 DPI because 600 DPI is typically the maximum for scanned documents. For the subsequent experiments, we used the model with 600 DPI training images.

B. IM2LATEX-100K

This Section presents the results on the *im2latex-100k* test set. *im2latex-100k* contains images with 100 DPI. However, *i2l-strips* and *i2l-pool* were trained with 200 DPI images, and our model was trained with 600 DPI images so they require larger images. We used two techniques to create larger images. First, we resized the original images with OpenCV to the training size. Second, we rendered the original MEs without normalization using a LaTeX environment to eliminate the influence of insufficient resolution.

As shown in Table 4, rendering an image with a higher resolution achieves better results as resizing the original images. *MathNet* achieved the same Edit scores (88.6%) as *WYGIWYS* with the resized images. However, the Edit error rate (1 - Edit score) nearly halved from 11.4% to 5.3% when the images were rendered with 600DPI. *i2l-strips* and *i2l-nopool* performed poorly with the resized images (32.5% and 32.0%), but similarly to *WYGIWYS* with the rendered images (86.9% and 86.8%). Interestingly, the exact match score of *MathNet* is low compared to the other systems. Hence, the fewer errors of *MathNet* must be more widely spread over the different MEs than those of the other systems.

TABLE 4. Results of the *im2latex-100k* test set. We run the models once with the original images, resized to the training size, and once with the images rendered with the optimal resolution.

Model	Edit [%]	Bleu-4 [%]	EM [%]
WYGIWYS	88.6	90.3	78.6
i2l-strips (resized)	32.5	12.7	0
i2l-strips (rendered)	86.9	86.1	76.3
i2l-nopool (resized)	32.0	13.4	0
i2l-nopool (rendered)	86.8	85.9	76.2
MathNet (our) (resized)	88.6	86.0	31.8
MathNet (our) (rendered)	94.7	94.5	63.4

C. IM2LATEXV2

This section presents the results with the *im2latexv2* test set. We assigned a random font for each ME in the test set. We used the same font-ME combination for all models to avoid influencing the results by using different fonts for the same ME. Since *im2latexv2* uses 600 DPI images, we resized the images for *i2l-strips*, *i2l-nopool*, and *WYGIWYS* to the training image resolution. As presented in Table 5, *WYGIWYS*'s Edit score drops dramatically from 88.6% to 37.2% compared to *im2latex-100k*. However, *i2l-strips* and *i2l-nopool* handle multiple fonts better, with only a small decrease of 11 pp. and 10.8 pp. in the Edit score. In contrast, our model shows a 2.5 pp. increase in the Edit score. We attribute this increase to the fact that *im2latex-100k* includes problematic mathematical fonts, as explained in Section VI-A4.

D. REALFORMULA

This section presents the results of the *realFormula* test set. Table 6 provides an overview of the results. The table shows that our model reaches an Edit score of 88.3%. This is about three times higher than *WYGIWYS* (27.5%) and approximately one-third higher than *i2l-strips* (65.1%) and *i2l-nopool* (65.2%). In order to quantify the impact of multi-line formulae we have split the MEs into multi-line (M) and single-line (S) MEs as discussed in Section VI-A5. To determine the influence of the array element we have filtered out all MEs with the token elements `\begin{array}` and `\end{array}` (nA), which is discussed in Section VI-A3. Additionally, we have filtered out all MEs with mathematical fonts (nMF); this issue is discussed in Section VI-A4.

E. INFYMDB-1

This Section presents the results on the *InfyMDB-1* test set [36]. *InfyMDB-1* contains 4400 images of scanned MEs with a resolution of 600 DPI. We used the pandoc library to convert the MathML GT into LaTeX GT and processed the resulting LaTeX strings similar to the other datasets.

As shown in Table 7, the resulting performance of *MathNet* is about the same compared to *realFormula* test set. However, it demonstrates that *MathNet* is not significantly affected by the noise of the scanning process. In contrast, the performance of *WYGIWYS*, *i2l-strips*, and *i2l-nopool* drops by 10.2 pp., 2.0 pp., and 1.6 pp.. This highlights that these models are probably affected by the noise of the scanning process. However, since our focus is on scientific PDFs, we

TABLE 5. Results of the *im2latexv2* test set. We resized the images to the optimal size. Errors is the summed Levenshtein Distance over all MEs. Array Errors is the summed Levenshtein Distance of all MEs with an array structure. nA is the Edit Score of all MEs without an array structure.

Model	Train Dataset	Edit [%]	Bleu-4 [%]	EM [%]	Errors	Array Errors	Edit nA [%]
WYGIWYS	im2latex-100k	37.2	23.9	0	564'700	31'742	37.1
I2l-strips	im2latex-140k	75.9	65.9	10.3	143'802	28'539	79.2
I2l-nopool	im2latex-140k	76.0	66.4	10.4	144'860	28'015	78.9
MathNet (our)	<i>im2latexv2</i>	97.2	96.8	83.9	16'596	8'728	98.6
MathNet (our)	im2latex-100k	78.2	65.9	10.3			
MathNet (our)	<i>im2latexv2</i> (vanilla font)	90.4	84.3	26.4			

TABLE 6. Edit scores [%] of the *realFormula* test set. S: single line ME, M: multi-line ME, nA: no arrays, A: arrays, nMF: no mathematical fonts, MF: mathematical fonts

Model	all	S	S nA	S A	S nMF	S MF	M
WYGIWYS	27.5	28.6	29.6	11.6	28.9	27.8	22.5
I2l-strips	65.1	76.6	81.7	17.7	82.6	65.1	15.9
I2l-nopool	65.2	77.1	81.9	20.7	83.5	65.0	13.9
MathNet (our)	88.3	92.5	93.3	84.1	94.1	89.5	71.2

TABLE 7. Prediction results on the InftyMDB-1 dataset (scanned MEs).

Model	Edit [%]	Bleu-4 [%]	EM [%]
WYGIWYS	17.3	7.1	0
I2l-strips	63.5	46.5	4.1
I2l-nopool	63.2	46.4	3.4
MathNet (our)	89.2	85.4	35.4

assume that scientific PDFs are usually available in native digital format. Hence, scanned documents with geometric deformation, coloration, and noise are considered as not in our research focus.

VI. DISCUSSION

A. DATA RELATED ACHIEVEMENTS AND CHALLENGES

As our results reveal, our data-centric approach with the LaTeX normalization and augmentation process is very beneficial for the training of robust printed MER models. The influence of our normalization and the use of multiple fonts on the model performance is discussed in Section VI-A1. Section VI-A2 demonstrates that our model is adept at working with fonts not included in the training set.

However, in our error analysis, we encountered two significant challenges. First, the array element was the main culprit of errors, as detailed in Section VI-A3. Second, the absence of mathematical fonts and multi-line MEs in the *im2latexv2* training dataset poses a challenge for our model on the *realFormula* test set, as discussed in Section VI-A4 and Section VI-A5. Section VI-A6 gives an overview of the most frequent token errors with MathNet and *im2latexv2*.

1) The Impact of Normalization and Multiple Fonts

We conducted experiments to separately analyse the influences of our model architecture, our normalization process, and the use of multiple fonts. We trained the model three times, once with the *im2latex-100k* dataset, once with the *im2latexv2* dataset using only the basic font, and once with the full *im2latexv2* dataset. The results are shown in Table 5. When we used the *im2latex-100k* dataset, our model's Edit

score (78.2%) was more than double that of WYGIWYS (37.2%) and was 2.3 pp. 2.2 pp. higher than *i2l-strips* and *i2l-nopool*. This demonstrates the beneficial network design of our model. The advantage of our model architecture is analyzed further in Section VI-B. However, the normalization process has a much stronger impact on the model's Edit score, with a 12.2 pp. improvement when using the *im2latexv2* dataset with the vanilla font for all MEs. The remaining 6.8 pp. improvement is explained by the use of multiple fonts for the MEs during training. In summary, the model architecture is marginally better as state of the art model architectures. However, two-thirds of the improvement compared to state of the art models are due to our LaTeX normalization process, while the remaining third is attributed to the use of multiple fonts. This reveals the significant influence of our LaTeX normalization process during model training and, hence, the value of the new dataset *im2latexv2*.

2) Non-Training Fonts

The *im2latexv2* training set only includes 30 of the 59 fonts in the test set. We tested our model's ability to work with fonts not in the training set and found that the font influence is negligible. Specifically, we achieved a 97.5% Edit score for MEs with fonts in the training set and 96.8% for MEs with fonts that were not. Overall, this demonstrates our model's strong generalizability.

3) Array Issue

LaTeX users normally use the array structure to create a matrix (`\begin{array} \cdots \end{array}`), but some authors use it to format their ME instead. To address this unwanted variation we introduced normalization steps in Section III to reduce the use of the array environment for formatting purposes. However, even with our normalization process, the array structure remains challenging for MER, as shown in Table 5. Out of all the prediction errors on the *im2latexv2* test set with our model, 52.6% are related to MEs with an array structure. However, this array structure is only present

TABLE 8. Prediction results for MEs in the *realFormula* test set with mathematical fonts.

Mathematical Font Type	Edit [%]	Bleu-4 [%]	EM [%]
<code>\boldsymbol</code>	98.4	96.4	63.6
<code>\mathcal</code>	90.4	83.7	18.2
<code>\mathbb</code>	84.6	75.4	0.0
<code>\operatorname</code>	83.4	75.7	0.0

in 4.8% of all the MEs. Therefore, by removing MEs that use the array structure, our model's Edit error rate is reduced by 50% (from 2.8% to 1.4%). *i2l-strips* and *i2l-nopool* also see reductions from 24.1% to 20.8% and from 24% to 21.1%, respectively. The effect on *WYGIWYS* is not significant. We attribute this to the high overall Edit error rate of *WYGIWYS*. The same problems with array structures can be seen in the results of Table 6 for the *realFormula* test set. For instance, *MathNet* achieves an Edit score of 93.3% for single line MEs without arrays and 84.1% for single line MEs with arrays.

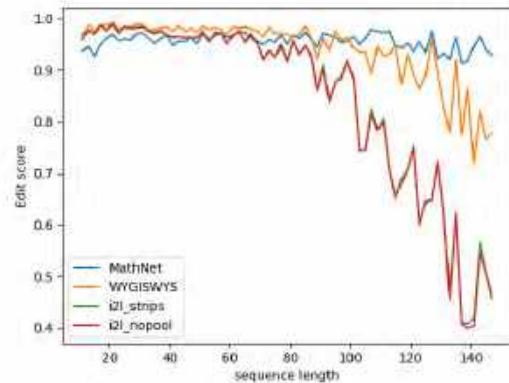
4) Mathematical Font Issue

In ME, changes to the font style of symbols (mathematical fonts) are used to indicate, e.g., vectors and spaces. As these mathematical fonts are not rendered correctly for all fonts, we decided to remove all mathematical font tokens in *im2latex2* to ensure the images are rendered correctly. Consequently, *MathNet* cannot detect mathematical fonts. Our results with the *realFormula* test set reveal that removing MEs with mathematical fonts in the training set has a significant influence on the model's real-world performance. Without counting the mathematical font tokens as an error, the Edit score of MEs without mathematical fonts is 94.1% (column S nMF), whereas MEs rendered with mathematical fonts, it drops to only 89.5% (column S MF). Table 8 shows that MEs rendered with the three mathematical fonts `\mathcal`, `\mathbb`, and `\operatorname` are especially challenging for *MathNet*. In contrast, the mathematical font `\boldsymbol` has no negative influence on the performance. Nevertheless, mathematical fonts are a limitation of *MathNet* and *im2latexv2* and, hence, the predicting results of *MathNet* deteriorate for MEs containing mathematical fonts. This issue is to be addressed in future research.

5) Multi-line ME

The MEs in the *im2latex-100k* dataset are limited to 150 tokens, so there are almost no multi-line MEs. However, in the *realFormula* dataset we had to drop 69 MEs because they were too large, and these were all multi-line MEs. Together with the 11 multi-line MEs in the final dataset, 80 of the original 200 MEs were multi-line MEs (see Table 2). This reveals that a real-world MER needs to handle multi-line MEs in addition to single-line MEs.

Table 6 shows that our model performs 30% better in terms of Edit score for single-line MEs (92.5%) compared to multi-line MEs (71.2%). This may be because our training set, *im2latexv2*, consists primarily of single-line expressions. The other models suffer a much more dramatic performance drop

**FIGURE 5.** The plot shows the average edit score per sequence length for the different models and the *im2latex-100k* dataset. The x-axis shows the number of tokens in the ME with a bin width of 3. The y-axis shows the average Edit score of each bin. A perfect prediction has a Edit score of 1.

for multi-line MEs to 14-23% Edit score. Furthermore, by using a straightforward y-cut algorithm, we can strongly improve our model's performance for multi-line MEs from 71.2% to 96.2% Edit score. As a result, when the y-cut algorithm performs well, we can robustly recognize multi-line MEs even with our model mainly trained on single-line MEs.

6) Most Frequent Token Errors

To better understand the open challenges of our *MathNet* model, we analyzed the Levenshtein operations needed to correct the predictions. Table 9 shows the 10 most frequent tokens that needed to be corrected. It is not surprising that the curly brackets are the primary culprit of errors because they are the most frequent tokens in the GT. Also, the sub- and superscript tokens (`_` and `^`) are still tricky for our model, even after our normalization step.

The replace operations reveal that the model is mainly confused by visually very similar symbols. However, their occurrences are small compared to the number of errors with curly brackets.

B. MODEL RELATED ACHIEVEMENTS AND CHALLENGES

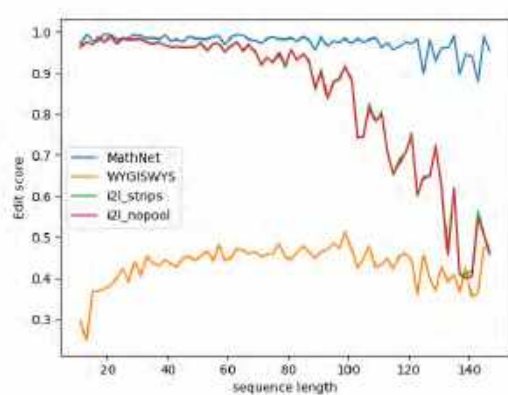
As discussed in Section II, many MER models employ LSTMs with specialized mechanisms to improve long-distance learning. We addressed this issue using a transformer architecture. Our analysis, depicted in Figures 5 and 6, shows that the Edit score of *MathNet* does not decrease with the sequence length of the MEs, indicating that transformers are effective in learning long-distance relationships in MEs.

VII. CONCLUSION

We introduced the novel printed MER model *MathNet*, incorporating a CvT encoder and transformer decoder. *MathNet* achieves outstanding results for *im2latex-100k* (Edit score:

TABLE 9. Analysis of the Levenshtein operations required to correct the *MathNet* predictions on the *im2latexv2* test set. The table shows the 10 most frequent tokens required to be inserted or deleted and the 10 most frequent pairs of tokens that must be replaced by the other.

Insert		Delete		Replace		Count
Token	Count	Token	Count	Token 1	Token 2	
}	2078	}	666	*	\ast	196
{	2000	{	589	^	\star	95
^	678	—	122	a	\alpha	79
—	678	*	96	\nu	v	74
2	345	^	63	\phi	\varphi	71
)	344	—	49	\rangle	>	42
—	271	1	47	\dots	\cdots	40
(	263	2	46	\epsilon	\varepsilon	36
1	242	&	43	\langle	<	34
,	196	c	42	\psi	\Psi	29

**FIGURE 6.** The plot shows the average edit score per sequence length for the different models and the *im2latexv2* dataset. The x-axis shows the number of tokens in the ME with a bin width of 3. The y-axis shows the average Edit score of each bin. A perfect prediction has a Edit score of 1.

94.7%), *im2latexv2* (Edit score: 97.2%), *realFormula* (Edit score: 88.3%), and *InftyMDB-1* (Edit score: 89.2%), reducing the Edit error rate to the prior state of the art for these datasets by 53.5% (from 11.4% to 5.3%), 88.3% (from 24% to 2.8%), 66.4% (from 34.8% to 11.7%), and 70.4% (from 36.5% to 10.8%), respectively. These results were achieved with our transformer-based model architecture and on an inherently data-centric approach normalizing and augmenting the training data. We found that detrimental variations in the LaTeX GT of *im2latex-100k* exist. To reduce this undesired variations, we proposed a LaTeX normalization method. Our LaTeX normalization process enables the model to focus on the canonical form of an ME instead of learning non-relevant variations. We demonstrated that our LaTeX normalization process is mainly responsible for the model's superior performance. Moreover, we introduced an augmented dataset, *im2latexv2*, an enhanced and normalized version of *im2latex-100k* with multiple fonts, and *realFormula* which contains annotated real ME images from arXiv papers. We also showed that a simple y-cut algorithm can expand single-line MER to multi-line MER.

Despite promising effectiveness, the Edit scores of all mod-

els investigated were significantly lower on *realFormula* and *InftyMDB-1* compared to *im2latex-100k* and *im2latexv2*, which indicates a difference between synthetic (*im2latex-100k* and *im2latexv2*) and real-world datasets (*realFormula* and *InftyMDB-1*). The removal of mathematical fonts styles in *im2latexv2*, such as bold and italics, limits the correct recognition of MEs that use these mathematical fonts styles in *realFormula*. An extended version of *im2latexv2* with mathematical fonts could solve this problem. Additionally, the correct cutting of ME lines heavily supports multi-line ME recognition, making stable line detectors a precondition.

After testing the handwritten benchmark dataset *CROHME* with our model *MathNet* and our LaTeX normalization, we could not find evidence that our LaTeX normalization process helps to improve the recognition performance. We think this is because the characteristics of *CROHME* and *im2latex-100k* are vastly different. The MEs in *CROHME* are on average only one-third as long as in *im2latex-100k*, and the vocabulary is significantly smaller, consisting of only 142 tokens compared to 500 tokens in printed MER. As a result, our LaTeX normalization only reduces the original 142 tokens to 121 (canonical) tokens, which is much less than with *im2latex-100k*. Furthermore, the MEs in *CROHME* are simpler and do not contain arrays, mathematical fonts, and other complex elements. This leads to the conclusion that the detrimental variation in *CROHME* is much lower than in *im2latex-100k*. However, we believe that for more complex handwritten MEs, our LaTeX normalization process could be as beneficial as it is for printed MER.

Generative pretrained transformers with multimodal input have shown significant progress in image recognition. However, testing a few ME images with GPT-4 from OpenAI indicates that the results, although impressive, have not yet reached the state of the art in MER. Nevertheless, combining generative AI with MER could be a promising approach worth exploring.

For our upcoming research steps, we plan to combine *FormulaNet* [32] and *MathNet* to develop a semi-automatic captioning system for MEs in PDFs. With this system, we expect to significantly improve the accessibility of PDFs specifically for MEs and also enable easy searching and extracting of MEs from PDFs.

REFERENCES

- [1] Felix M. Schmitt-Koopmann, Elaine M. Huang, and Alireza Darvishy. Accessible PDFs: Applying Artificial Intelligence for Automated Remediation of STEM PDFs. In *Proceedings of the 24th International ACM SIGACCESS Conference on Computers and Accessibility, ASSETS '22*, pages 1–6, New York, NY, USA, 2022. Association for Computing Machinery. doi: 10.1145/3517428.3550407.
- [2] Abdelwaheb Belaid and Jean-Paul Haton. A Syntactic Approach for Handwritten Mathematical Formula Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(1):105–111, January 1984. doi: 10.1109/TPAMI.1984.4767483.
- [3] Kam-Fai Chan and Dit-Yan Yeung. Mathematical expression recognition: a survey. *International Journal on Document Analysis and Recognition*, 3(1):3–15, August 2000.
- [4] Ridhi Aggarwal, Shilpa Pandey, Anil Kumar Tiwari, and Gaurav Harit. Survey of Mathematical Expression Recognition for Printed and Handwritten Documents. *IETE Technical Review*, 39(6):1245–1253, November 2022. doi: 10.1080/02564602.2021.2008277.
- [5] Anssi Kanervisto. im2latex-100k, June 2016. doi: 10.5281/zenodo.56198.
- [6] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, February 2021. doi: 10.1145/3446776.
- [7] Devansh Arpit, Stanislaw Jastrzebski, Nicolas Ballas, David Krueger, Emanuel Bengio, Maxinder S. Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, and Simon Lacoste-Julien. A Closer Look at Memorization in Deep Networks. In *Proceedings of the 34th International Conference on Machine Learning*, pages 233–242. PMLR, July 2017.
- [8] DeepLearningAI. A Chat with Andrew on MLOps: From Model-centric to Data-centric AI, March 2021. [Online]. Available: <https://www.youtube.com/watch?v=06-AZXmwHjo>.
- [9] Paul-Philipp Luley, Jan M. Deriu, Peng Yan, Gerrit A. Schatte, and Thilo Stadelmann. From Concept to Implementation: The Data-Centric Development Process for AI in Industry. In *2023 10th IEEE Swiss Conference on Data Science (SDS)*, pages 73–76, June 2023. doi: 10.1109/SDS57534.2023.00017.
- [10] Thilo Stadelmann, Tino Klamt, and Philipp H. Merkt. Data Centrism and the Core of Data Science as a Scientific Discipline. *Archives of Data Science, Series A*, 8(2):16, 2022. doi: 10.5445/IR/1000143637.
- [11] Robert H. Anderson. Syntax-directed recognition of hand-printed two-dimensional mathematics. In *Symposium on Interactive Systems for Experimental Applied Mathematics: Proceedings of the Association for Computing Machinery Inc. Symposium*, pages 436–459, New York, NY, USA, August 1967. Association for Computing Machinery. doi: 10.1145/2402536.2402585.
- [12] Yejing Xie, Harold Mouchère, Foteini Simistira Liwicki, Sumit Rakesh, Rajkumar Saini, Masaki Nakagawa, Cuong Tuan Nguyen, and Thanh-Nghia Truong. ICDAR 2023 CROHME: Competition on Recognition of Handwritten Mathematical Expressions. In Gernot A. Fink, Rajiv Jain, Koichi Kise, and Richard Zanibbi, editors, *Document Analysis and Recognition - ICDAR 2023*, Lecture Notes in Computer Science, pages 553–565, Cham, 2023. Springer Nature Switzerland. doi: 10.1007/978-3-031-41679-8_33.
- [13] Masakazu Suzuki, Fumikazu Tamari, Ryoji Fukuda, Seiichi Uchida, and Toshihiro Kanahori. INFITY: an integrated OCR system for mathematical documents. In *Proceedings of the 2003 ACM symposium on Document engineering, DocEng '03*, pages 95–104, New York, NY, USA, November 2003. Association for Computing Machinery. doi: 10.1145/958220.958239.
- [14] Christopher Malon, Seiichi Uchida, and Masakazu Suzuki. Mathematical symbol recognition with support vector machines. *Pattern Recognition Letters*, 29(9):1326–1332, July 2008. doi: 10.1016/j.patrec.2008.02.005.
- [15] Yuntian Deng, Anssi Kanervisto, Jeffrey Ling, and Alexander M. Rush. Image-to-Markup Generation with Coarse-to-Fine Attention. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70, ICML '17*, pages 980–989, Sydney, NSW, Australia, August 2017. JMLR.org.
- [16] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, November 1997. doi: 10.1162/neco.1997.9.8.1735.
- [17] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1179.
- [18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30, pages 1–11. Curran Associates, Inc., 2017.
- [19] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/W14-4012.
- [20] Xiaohang Bian, Bo Qin, Xiaozhe Xin, Jianwu Li, Xuefeng Su, and Yanfeng Wang. Handwritten Mathematical Expression Recognition via Attention Aggregation Based Bi-directional Mutual Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(1):113–121, June 2022. doi: 10.1609/aaai.v36i1.19885.
- [21] Bohan Li, Ye Yuan, Dingkan Liang, Xiao Liu, Zhilong Ji, Jinfeng Bai, Wenyu Liu, and Xiang Bai. When Counting Meets HMER: Counting-Aware Network for Handwritten Mathematical Expression Recognition. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVIII*, pages 197–214, Berlin, Heidelberg, 2022. Springer-Verlag. doi: 10.1007/978-3-031-19815-1_12.
- [22] Zuoyu Yan, Xiaode Zhang, Liangcai Gao, Ke Yuan, and Zhi Tang. ConvMath: A Convolutional Sequence Network for Mathematical Expression Recognition. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 4566–4572, Milan, Italy, January 2021. IEEE. doi: 10.1109/ICPR48806.2021.9412913.
- [23] Abolfazl Mirkazemy, Peyman Adibi, Seyed Mohammad Saied Ehsani, Alireza Darvishy, and Hans-Peter Hutter. Mathematical expression recognition using a new deep neural model. *Neural Networks*, 167:865–874, October 2023. doi: 10.1016/j.neunet.2023.08.045.
- [24] Jian Wang, Yunchuan Sun, and Shenling Wang. Image To Latex with DenseNet Encoder and Joint Attention. *Procedia Computer Science*, 147:374–380, 2019. doi: 10.1016/j.procs.2019.01.246.
- [25] Zhe Li, Lianwen Jin, Songxuan Lai, and Yecheng Zhu. Improving Attention-Based Handwritten Mathematical Expression Recognition with Scale Augmentation and Drop Attention. In *2020 17th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 175–180. IEEE Computer Society, September 2020. doi: 10.1109/ICFHR2020.2020.00041.
- [26] Shuai Peng, Liangcai Gao, Ke Yuan, and Zhi Tang. Image to LaTeX with Graph Neural Network for Mathematical Formula Recognition. In *Document Analysis and Recognition – ICDAR 2021: 16th International Conference, Lausanne, Switzerland, September 5–10, 2021, Proceedings, Part II*, pages 648–663, Berlin, Heidelberg, September 2021. Springer-Verlag. doi: 10.1007/978-3-030-86331-9_42.
- [27] Mahshad Mahdavi, Richard Zanibbi, Harold Mouchère, Christian Viard-Gaudin, and Utpal Garain. ICDAR 2019 CROHME + TFD: Competition on Recognition of Handwritten Mathematical Expressions and Typeset Formula Detection. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1533–1538, September 2019. doi: 10.1109/ICDAR.2019.00247.
- [28] Sumeet S. Singh. Teaching Machines to Code: Neural Markup Generation with Visual Attention, June 2018. doi: 10.48550/arXiv.1802.05415.
- [29] Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. Nougat: Neural Optical Understanding for Academic Documents, August 2023. doi: 10.48550/arXiv.2308.13418.
- [30] Niclas Simmler, Pascal Sager, Philipp Andermatt, Ricardo Chavarriaga, Frank-Peter Schilling, Matthias Rosenthal, and Thilo Stadelmann. A Survey of Un-, Weakly-, and Semi-Supervised Learning Methods for Noisy, Missing and Partial Labels in Industrial Vision Applications. In *2021 8th Swiss Conference on Data Science (SDS)*, pages 26–31, June 2021.
- [31] Felix Schmitt-Koopmann. MathNet: A Data-Centric Approach for Printed Mathematical Expression Recognition., 2023. [Online]. Available: <https://github.com/felix-schmitt/MathNet>.
- [32] Felix M. Schmitt-Koopmann, Elaine M. Huang, Hans-Peter Hutter, Thilo Stadelmann, and Alireza Darvishy. FormulaNet: A Benchmark Dataset for Mathematical Formula Detection. *IEEE Access*, 10:91588–91596, 2022. doi: 10.1109/ACCESS.2022.3202639.
- [33] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias

- Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, pages 1–21. OpenReview.net, 2021.
- [34] Haipeng Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. CvT: Introducing Convolutions to Vision Transformers. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22–31, October 2021. doi: 10.1109/ICCV48922.2021.00009.
- [35] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *CoRR*, pages 1–13. Ithaca, NY: arXiv.org, December 2014. doi: <https://hdl.handle.net/11245/1.505367>.
- [36] Akio Fujiyoshi, Masakazu Suzuki, and Seiichi Uchida. InftyMDB-1, 2009. [Online]. Available: <https://www.inftyproject.org/download/inftydb/InftyMDB-1.zip>.



THILO STADELMANN is professor of AI/ML at the ZHAW School of Engineering in Winterthur, Switzerland, director of the ZHAW Centre for Artificial Intelligence and head of its Machine Perception and Cognition Group. He studied computer science in Giessen and Marburg and received his Doctor of Science degree from Marburg University, Germany, in 2010, where he worked on multimedia analysis and voice recognition. Thilo held engineering and leadership roles in the automotive industry for several years prior to his appointment at the ZHAW. His current research focuses on robust deep learning to solve diverse pattern recognition tasks such as document analysis or industrial or medical computer vision.



FELIX M. SCHMITT-KOOPMANN received the B.Sc. degree in mechanical engineering and the M.Sc. degree in Robotics, Systems, and Control from ETH Zurich, in 2019 and 2021. He is currently a Ph.D. Student with the People and Computing Lab of the University of Zurich, and he is a member of the Institute of Applied Informatics of ZHAW. His research interests include accessibility, AI, and document analysis.



ELAINE M. HUANG is a professor of Human-Computer Interaction in the Department of Informatics at the University of Zurich, where she leads the People and Computing Research Group. Prior to joining UZH in 2010, she was a researcher at Motorola Labs and a professor in the department of Computer Science at the University of Calgary. She received her PhD from the College of Computing at the Georgia Institute of Technology in 2006. Her research focuses on the use of technology to address issues of inequality and other societal challenges.



ALIREZA DARVISHY is professor for ICT Accessibility and head of the ICT Accessibility Lab at Zurich University of Applied Sciences in Switzerland. He serves an independent reviewer for European research projects such as the Active Assisted Living (AAL) program, and is principle investigator of the “Accessible Scientific PDFs for All” project, funded by the Swiss National Science Foundation.

...



HANS-PETER HUTTER studied Electrical Engineering at ETH Zurich where he received a doctor of technical science degree in 1997 for his work on hybrid HMM/ANN approaches to speech recognition over telephone lines. Hans-Peter joined the UBS Ubilab as post doc where he worked on a European project for HMM-based speaker identification over the telephone. At the same time he was co-lecturer at ETHZ in two speech processing modules. In 1997 he joined the ZHAW Zurich University of Applied Sciences in Winterthur where he worked as a professor in computer science on various projects in the area of speech recognition and user centered design of graphical and voice user interfaces. In 2005 he founded the InIT Institute of applied Information Technology at the ZHAW School of Engineering together with his colleagues and was head of the institute until 2010. At the same time, he was also head of the Human-Information Interaction group of the InIT which he is still leading today.

Originally published in Proceedings of the 11th IEEE Swiss Conference on Data Science (SDS'24), Zurich, Switzerland, May 31, 2024, DOI 10.1109/SDS60720.2024.00040.

ScalaGrad: A Statically Typed Automatic Differentiation Library for Safer Data Science

Benjamin Meyer^{*†}, Thilo Stadelmann^{*§}, Marcel Lüthi[‡]
mebr@zhaw.ch, stdm@zhaw.ch, marcel.luethi@unibas.ch

^{*}Centre for Artificial Intelligence, ZHAW School of Engineering, Winterthur, Switzerland

[†]Institute of Neuroinformatics, ETH & University of Zurich, Zurich, Switzerland

[‡]Departement Mathematik und Informatik, University of Basel, Basel, Switzerland

[§]European Centre for Living Technology (ECLT), Ca' Bottacin, Venice, Italy

[§]Fellow, ECLT (European Centre for Living Technology, Venice, Italy) and Senior Member, IEEE

Abstract—While the data science ecosystem is dominated by programming languages that do not feature a strong type system, it is widely agreed that using strongly typed programming languages leads to more maintainable and less error-prone code and ultimately more trustworthy results. We believe Scala 3 would be an excellent contender for data science in a strongly typed language, but it lacks a general automatic differentiation library, e.g., for gradient-based learning. We present ScalaGrad, a general and type-safe automatic differentiation library designed for Scala. It builds on and improves a novel approach from the functional programming community using immutable duals, which is conceptually simple, asymptotically optimal and allows differentiation of higher-order code. We demonstrate the ease of use, robust performance, and versatility of ScalaGrad through its applications to deep learning, higher-order optimization, and gradient-based sampling. Specifically, we show an execution speed comparable to PyTorch for a simple deep learning use case, capabilities for higher-order differentiation, and opportunities to design more specialized libraries decoupled from ScalaGrad. As data science challenges evolve in complexity, ScalaGrad provides a pathway to harness the inherent advantages of strongly typed languages, ensuring both robustness and maintainability.

Index Terms—Automatic Differentiation, Scala 3, ScalaGrad

I. INTRODUCTION

The Python programming language dominates the field of data science [1]. Python is easy to learn, fast to experiment with, and has a vast ecosystem of libraries like scikit-learn [2], TensorFlow [3], PyTorch [4] and others [5]–[7]. Despite Python's strengths, it has a dynamic type system that hinders maintainability and trustworthiness in complex applications. With the growing complexity of data science challenges, the case for statically typed languages becomes compelling.

Scala 3 has a static, powerful, and sound type system [8]. It propagates a functional programming paradigm without losing the flexibility for side effects if needed, for example, mutable state, which can be necessary for performance optimization. Scala 3's concise and braceless syntax allows for fast experimentation and eases the switch from Python. Scala has broad tool support, as it is an established language for distributed systems and big data applications. However, Scala 3 lacks a general automatic differentiation library, e.g., to facilitate ubiquitous gradient-based learning [9].

In this paper, we introduce ScalaGrad, a general and type-safe automatic differentiation library written in and designed for Scala. ScalaGrad builds on a novel approach introduced in the functional programming community by Krawiec et al. [10], using immutable duals, resulting in easy-to-reason code while being asymptotically efficient. Moreover, ScalaGrad improves on this approach by removing the necessity of a state monad, making the approach viable for Scala and allowing straightforward higher-order differentiation support. ScalaGrad is easy to use as it can differentiate over general control flow statements like if-else statements, loops, monads, or exceptions. It supports different automatic differentiation modes and has robust performance by hardware-accelerated linear algebra operations using the Breeze library [11]. Demonstrating ScalaGrad's versatility, we applied it to deep learning, higher-order optimization, and gradient-based sampling. Specifically, the deep learning application demonstrates ScalaGrad's performance, calculating 28,630 partial derivatives 938 times in around 5 seconds. The higher-order optimization application demonstrates ScalaGrad's higher-order differentiation capabilities. And the gradient-based sampling application indicates the unique advantage of ScalaGrad's *type-safe* differentiation, as the algorithm was implemented decoupled from ScalaGrad while maintaining type safety. ScalaGrad and all applications have been open-sourced at <https://github.com/benikm91/scalagrad>.

As automatic differentiation is pivotal for gradient-based applications, ScalaGrad represents a significant step towards realizing a vision of type-safe data science in the Scala ecosystem.

II. RELATED WORK

Automatic Differentiation. Automatic differentiation usually runs in one of two modes, the forward mode or the reverse mode, with the selection dependent on the structure of the function to differentiate. The forward mode was discovered independently multiple times in the 50s, 60s, and 70s [12]–[16], and the reverse mode was discovered independently multiple times in the 70s and 80s [17]. In recent years, researchers have expanded the theoretical groundwork for automatic differentiation. Elliott [18] highlights that derivatives

```
def square(alg: MatrixAlgebraDSL)(x: alg.Scalar): alg.Scalar = x * x // Define function to derive
import ForwardMode.derive as d // Import derive function from ScalaGrad
val dSquare = d(square) // Derive function
assert(dSquare(BreezeDoubleMatrixAlgebraDSL)(3.0) == 2.0 * 3.0) // Apply derived function and check result
```

Fig. 1: Minimal example of using ScalaGrad. First, the `square` function is defined using `MatrixAlgebraDSL` to abstract over the multiplication operation. Next, the function is differentiated using ScalaGrad's forward mode, resulting in a new function, `dSquare`, the derivative of `square`.

are linear maps and reduces automatic differentiation to two basic operations: the fork and the chain operation. Based on this idea, Krawiec et al. [10] show an elegant and *immutable* implementation for the reverse mode. The crucial new idea is to use a dual, often used for the forward mode, to track the chain and fork operations in the reverse mode. ScalaGrad builds on those concepts and adds *type-safety* to the automatic differentiation operation using Scala 3's match types [19] to compute the function's derivative type at compile-time. Additionally, ScalaGrad simplifies those concepts, making it straightforward to implement the differentiation mechanism in a differentiable way (an idea taken from JAX [20]), enabling higher-order differentiation.

Python ecosystem. In the Python ecosystem, there exist some automatic differentiation libraries. TensorFlow and PyTorch come bundled with a specialized automatic differentiation engine with syntax narrowly tailored to deep learning: One creates a tensor expression, marks the input tensors as such (the weights), and starts to differentiate from an output tensor (the loss). Over the years, TensorFlow and PyTorch added support for general automatic differentiation, including multiple outputs, the forward mode, and higher-order differentiation. However, the afterthought for general automatic differentiation lead to an overly complex syntax for those more advanced automatic differentiation cases. In recent years, libraries like JAX [21] and funtorch [22] have provided a cleaner, more functional syntax. Rather than a tensor expression, one defines a function on tensors and differentiates it by a higher-order function. This idea leads to a natural syntax for multiple outputs, different modes, and higher-order differentiation. Different modes are simply different higher-order functions. Higher-order differentiation is just the chaining of such function calls.

Python 3 supports type hints to improve maintainability and safety when programming [23]. Type hints are optional type annotations that can be used for static code analysis. However, there is no compilation phase enforcing the correctness of types, which in practice leads to code that rarely type-checks [24], and many Python libraries do not use type hints. Disregarding those practical concerns, Python 3's type hinting is additionally limited in expressive power compared to Scala 3's. Python does not support, e.g., singleton types, higher-kinded types, path-dependent types, or match types.

Scala ecosystem. In Scala, there exist automatic differentiation engines bundled with deep learning libraries [25], [26] as well as bindings to TensorFlow [27] and LibTorch [28], [29], making automatic differentiation possible in those domains. However, Scala has no general, stand-alone automatic differ-

entiation library; therefore, it lacks a fundamental building block for applications using derivatives, like deep learning and probabilistic programming with gradient-based sampling.

III. SCALAGRAD

We present ScalaGrad, a general automatic differentiation library written in Scala for Scala, built on and improving a novel approach from Krawiec et al. [10]. ScalaGrad supports higher-order differentiation, the forward and reverse mode, and can differentiate over any control flow statements. Numerical operations must be polymorphic to add automatic differentiation capabilities, which ScalaGrad achieves using its own type class. When differentiating a function, ScalaGrad tracks the resulting derivative's type, adding type safety to the automatic differentiation operation.

Linear algebra polymorphism. ScalaGrad must abstract over numerical operations like addition and multiplication to track the derivatives while executing those operations. Additionally, for performance reasons, ScalaGrad abstracts over basic linear operations to implement them efficiently by native routines (using Breeze [11]) while tracking the derivatives. For this, ScalaGrad defines a type class `MatrixAlgebra` with four types: a scalar, a column vector, a row vector, and a matrix. The type class defines operations on those types. Using four types adds some type safety, e.g., the undefined inner product between two column vectors does not exist, rather than throwing a runtime error. Finally, for a more concise syntax, ScalaGrad provides the domain-specific language `MatrixAlgebraDSL` using path-dependent types and wrapping the `MatrixAlgebra` type class. Figure 1 shows a minimal ScalaGrad example using the `MatrixAlgebraDSL` abstraction and the `BreezeDoubleMatrixAlgebraDSL` implementation. `BreezeDoubleMatrixAlgebraDSL` implements the `MatrixAlgebra` operations with the Breeze library with double precision. Implementing linear operations by native routines gives ScalaGrad a robust performance, as discussed in the deep learning application.

Efficient, immutable reverse mode implementation. In the reverse mode, the partial derivatives for a function f are calculated from each output to all inputs. So, first, the operations must be tracked while executing f from the inputs to the outputs, called taping, and then the tape must be run back in reverse order from each output. An essential idea of Krawiec et al. [10] is that it is unnecessary to tape all kinds of operations, but only the chain and the fork operations, representing scaling and addition. The idea works because derivatives are linear maps, and non-linear scaling factors can be eagerly calculated while taping [18]. For an efficient reverse

```

def add(alg: MatrixAlgebraDSL)(x1: alg.Scalar, x2: alg.Scalar): alg.Scalar = x1 + x2 // Define function add
import ForwardMode.derive as d // Import derive function from ScalaGrad
val dAdd = d(add) // Derive function add
val (dx1, dx2) = dAdd(BreezeDoubleMatrixAlgebraDSL)(3.0, 4.0) // Compiler knows that dAdd returns two scalars

```

Fig. 2: ScalaGrad tracks the type of the derivative of a function during differentiation. Here, the compiler knows that `dAdd` returns two scalars because `add` takes two scalar parameters and returns one scalar, so there are two partial derivatives.

mode implementation, one must run back each operation of the tape exactly once. Therefore, all effects on dependent operations must be accumulated before running back that operation. Common automatic differentiation libraries ensure complete accumulation using global state. They track the operations in the *exact order* of execution using, for example, a global counter as an ordering number while taping and then they run the tape back in the reverse order of execution. The reverse order of execution ensures that all dependent effects are run back and accumulated before running back an operation. ScalaGrad simplifies this approach: it ensures complete accumulation without global state. ScalaGrad tracks the *partial order* of operations' dependencies rather than the order of execution. An ordering number for each operation tracks this partial order; the ordering number is just the maximum of all dependencies' ordering numbers plus one. Running back the tape while respecting the reverse partial order also ensures that all its dependent effects are first run back and accumulated before running back an operation. Figure 3 depicts the simple yet impactful insight behind this simplification.

<code>f(x1, x2) =</code>	Exact execution order and run back
<code> a1 = x1 * x2</code>	$a_1 \rightarrow a_2 \rightarrow b_1 \rightarrow b_2 \rightarrow y$
<code> a2 = x1 + x2</code>	$a_1 \leftarrow a_2 \leftarrow b_1 \leftarrow b_2 \leftarrow y$
<code> b1 = a1 * a2</code>	
<code> b2 = a1 + a2</code>	Partial dependencies order and run back
<code> y = b1 * b2</code>	$a_1, a_2 \rightarrow b_1, b_2 \rightarrow y$
<code>return y</code>	$a_1 \leftarrow a_2 \leftarrow b_2 \leftarrow b_1 \leftarrow y$

Fig. 3: Showcase the insight driving ScalaGrad's simplification. The left shows the example operations. The top right shows the commonly used exact execution order and how it is run back. The bottom right shows ScalaGrad's partial dependency order. There are multiple ways to run it back, here we just show one example. Both order ensure that all dependent effects have always been run back, before running back an operation.

Krawiec et al. [10] use a state monad to track the exact order of executions. An equivalent implementation in Scala is impractical to use, as every numeric operation would have to run inside this state monad's context¹. Utilizing mutable global state eliminates the need for a monad, however, it introduces complexities related to state corruption, e.g., re-entrance issues in higher-order differentiation and race conditions in multi-threaded applications. ScalaGrad's simplification allows the best of both worlds, it stays immutable while being ease to use. Additionally, higher-order differentiation, not supported by Krawiec et al. [10] mostly due to the state monad, can nat-

¹Krawiec et al. [10] use a sequence to sequence translation on a custom and limited language not having this issue.

TABLE I: Examples of type-safe differentiation in ScalaGrad

f input type	f output type	Resulting ∇f output type
(S, S)	M	(M, M)
(S, S)	(S, S)	$((S, S), (S, S))$
(S, CV)	(S, RV)	$((S, RV), (CV, M))$

S stands for Scalar, CV for ColumnVector, RV for RowVector, M for Matrix.

urally be supported. The higher-order optimization application demonstrates higher-order differentiation in ScalaGrad.

Type-safe differentiation. The return type of ∇f , the derivative of f , varies depending on f since ∇f returns all the partial derivatives. ScalaGrad captures this mathematical concept at the type level. Given a Scala function f of type $(\text{Scalar}, \text{Scalar}) \Rightarrow \text{Scalar}$, its derivative $d(f)$ has the type $(\text{Scalar}, \text{Scalar}) \Rightarrow (\text{Scalar}, \text{Scalar})$. ScalaGrad computes the type of $d(f)$ at *compile-time* based on f utilizing Scala 3's match types [19]. Figure 2 illustrates this concept in code. ScalaGrad can compute the type of $d(f)$ for any function f of scalars, vectors, and matrices; Table I shows some examples. Keeping track of the type during differentiation adds type safety to the automatic differentiation operation, adding robustness and maintainability to automatic differentiation applications. Additionally, it allows algorithms using derivatives to abstract over their function types, decoupling them from ScalaGrad but still being type-safe, as shown in the gradient-based sampling application.

IV. APPLICATIONS OF SCALAGRAD

To highlight ScalaGrad's versatility, we applied it to deep learning, higher-order optimization, and gradient-based sampling. The complete code for those applications can be found in the ScalaGrad repository.

Deep Learning. To showcase the robust performance of ScalaGrad, we trained a fully connected neural network classifying images from the MNIST dataset [9] by implementing the mini-batch gradient descent algorithm. The neural network has one hidden layer with 36 units, resulting in 28,630 parameters to learn, or, put differently, there are 28,630 partial derivatives to calculate each iteration. For one epoch consisting of 938 iterations (60,000 images with 64 as batch size), ScalaGrad took 5.18 ± 0.98 seconds measured on a MacBook M1 Pro and 100 epochs using OpenJDK 20. A comparable setup in PyTorch, with the same hardware, network architecture, hyperparameters, and floating precision, took 2.08 ± 0.19 seconds measured on 100 epochs using Python 3.9.

Higher-order optimization. To showcase the higher-order differentiation capabilities of ScalaGrad, we implemented Newton's method to train a linear model where $\nabla^2 f$, the second-order derivative, is needed.

ScalaGrad's differentiation mechanism inside the `derive` function is implemented with its own `MatrixAlgebra` abstraction, making the mechanism itself differentiable. Therefore, to get the second-order derivative of `f`, `derive` (here imported as `d`) is simply applied twice `d(d(f))`.

Gradient-based sampling. This application illustrates a unique design space for gradient-based applications using a *type-safe* automatic differentiation library. We implemented the Hamiltonian Monte Carlo algorithm by taking the posterior's derivative as a parameter. The derivative is just a Scala function and has no dependence on ScalaGrad. When passing the derivative computed by ScalaGrad to the algorithm, the compiler checks that the argument matches the parameter type, so this design is type-safe. However, another automatic differentiation library, or even a manually coded derivative could have been used instead, highlighting that the algorithm's implementation is decoupled from ScalaGrad.

V. DISCUSSION AND CONCLUSION

ScalaGrad is a general automatic differentiation library applicable to various use cases. However, there are certain limitations. Currently, ScalaGrad's `MatrixAlgebra` type class only abstracts operations of tensors with two or fewer dimensions. Additionally, there is no mechanism to extend new operations from the outside, which is necessary to bind other highly optimized native routines like hardware-accelerated activation functions in a deep learning library. Those limitations are not inherent to the method and can be solved by engineering efforts. Regarding future work, an obvious next step is to build a data science library on top of ScalaGrad, e.g., a deep learning or probabilistic programming library. Another future direction is to make ScalaGrad's abstraction more type-safe. One straightforward way is to use Scala's singleton types to track the dimensionality of tensors at compile-time to check for undefined operations. A more ambitious direction is to drop the notation of a tensor from the abstraction, as it is the unit of execution, not abstraction. For example, rather than having a 4-dimensional tensor representing a batch of images, one could have a list of an `Image` type. An open question is how to map operations on those abstractions back to tensor operations for execution with an acceptable loss of performance.

We presented ScalaGrad, a general automatic differentiation library for Scala. ScalaGrad can differentiate general Scala code, only requiring numerical operations to be implemented by its type class. It provides asymptotically efficient implementations for the forward and reverse mode, supports native linear operations, and tracks types while differentiating, adding type safety to the automatic differentiation operation. Additionally, we discussed our improved automatic differentiation implementation, removing the necessity for global state, which allows for an immutable and ease to use implementation as well as a natural higher-order differentiation support. Finally, we demonstrated ScalaGrad's versatility, robust performance, and type safety by applying it to deep learning, higher-order optimization, and gradient-based sampling.

REFERENCES

- [1] M. Braschler, T. Stadelmann, and K. Stockinger, "Data science," in *Applied Data Science: Lessons Learned for the Data-Driven Business*. Springer, 2019, pp. 17–29.
- [2] F. Pedregosa, G. Varoquaux *et al.*, "Scikit-learn: Machine Learning in Python," *J Mach Learn Res*, vol. 12, pp. 2825–2830, 2011.
- [3] M. Abadi, A. Agarwal *et al.*, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems," 2015.
- [4] A. Paszke, S. Gross *et al.*, "Automatic differentiation in PyTorch," in *NIPS 2017 Workshop on Autodiff*, 2017. URL: <https://openreview.net/forum?id=BJJsmfCZ>
- [5] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. ACM, 2016, pp. 785–794. URL: <https://doi.acm.org/10.1145/2939672.2939785>
- [6] G. Ke, Q. Meng *et al.*, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," *Advances in NIPS*, vol. 30, pp. 3146–3154, 2017.
- [7] E. Bingham, J. P. Chen *et al.*, "Pyro: Deep Universal Probabilistic Programming," *J. Mach. Learn. Res.*, vol. 20, pp. 28:1–28:6, 2019.
- [8] N. Amin, S. Grütter *et al.*, "The Essence of Dependent Object Types". Springer International Publishing, 2016, pp. 249–272.
- [9] Y. Lecun, L. Bottou *et al.*, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [10] F. Krawiec, S. Peyton Jones *et al.*, "Provably Correct, Asymptotically Efficient, Higher-Order Reverse-Mode Automatic Differentiation," *Proc. ACM Program. Lang.*, vol. 6, Jan. 2022. URL: <https://doi.org/10.1145/3498710>
- [11] D. Hall, "Breeze, Scala library for numerical processing, machine learning, and natural language processing," 2023. URL: <https://github.com/scalanlp/breeze>
- [12] L. Beda, L. Korolev *et al.*, "Programs for automatic differentiation for the machine BESM," Technical Report, Institute for Precise Mechanics and Computation Techniques, Tech. Rep., 1959.
- [13] R. E. Wengert, "A simple automatic derivative evaluation program," *Communications of the ACM*, vol. 7, pp. 463–464, 1964.
- [14] J. Joss, "Algorithmisches Differenzieren," Ph.D. dissertation, ETH Zurich, 1976.
- [15] G. Kedem, "Automatic Differentiation of Computer Programs," *ACM Trans. Math. Softw.*, vol. 6, no. 2, p. 150–165, Jun. 1980. URL: <https://doi.org/10.1145/355887.355890>
- [16] D. D. Warner, "A partial derivative generator." Bell Telephone Laboratories, Computing Science Technical Report No. 28, 1975.
- [17] A. Griewank, "Who invented the reverse mode of differentiation?" *Documenta Mathematica*, Jan. 2012.
- [18] C. Elliott, "The simple essence of automatic differentiation," Apr. 2018.
- [19] O. Blanvillain, J. I. Brachthäuser *et al.*, "Type-Level Programming with Match Types," *Proc. ACM Program. Lang.*, vol. 6, Jan. 2022. URL: <https://doi.org/10.1145/3498698>
- [20] JAX authors. Jax from scratch. Accessed: 18 August 2023. URL: <https://jax.readthedocs.io/en/latest/autodidax.html>
- [21] J. Bradbury, R. Frostig *et al.*, "JAX: composable transformations of Python+NumPy programs," 2018. URL: <https://github.com/google/jax>
- [22] R. Z. Horace He, "functorch: JAX-like composable function transforms for PyTorch," 2021. URL: <https://github.com/pytorch/functorch>
- [23] G. Van Rossum, J. Lehtosalo, and L. Langa, "PEP 484 - Type Hints." URL: <https://peps.python.org/pep-0484/>
- [24] I. Rak-amnoui, D. McCrean *et al.*, "Python 3 Types in the Wild: A Tale of Two Type Systems," in *Proceedings of the 16th ACM SIGPLAN International Symposium on Dynamic Languages*, ser. DLS 2020. Association for Computing Machinery, 2020, p. 57–70.
- [25] ThoughtWorks, "DeepLearning.scala: A simple library for creating complex neural networks," 2023. URL: <https://github.com/ThoughtWorksInc/DeepLearning.scala>
- [26] G. Modena, "Scalagrad: A tiny reverse-mode autodiff and neural network library," <https://github.com/gmodena/scalagrad>, 2020.
- [27] E. A. Platanios, "TensorFlow Scala," 2018. URL: https://github.com/eaplatanos/tensorflow_scala
- [28] Microsoft, "Scala bindings for LibTorch," 2023. URL: https://github.com/microsoft/scala_torch
- [29] S. Brunk, "Storch: Gpu accelerated deep learning and numeric computing for scala 3," <https://github.com/sbrunk/storch>, 2023.

Originally published in *Pattern Recognition Letters* 181, pp. 64-49, Elsevier, May 2024, DOI 10.1016/j.patrec.2024.03.016.



Pattern Recognition Letters
journal homepage: www.elsevier.com

Deep Neural Networks for Automatic Speaker Recognition Do Not Learn Supra-Segmental Temporal Features

Daniel Neururer^{a,2}, Volker Dellwo^b, Thilo Stadelmann^{a,c,2,**}

^aCentre for Artificial Intelligence, Zurich University of Applied Sciences, Technikumstrasse 71, 8400, Winterthur, Switzerland

^bDepartment of Computational Linguistics, University of Zurich, Andreasstrasse 15, 8050, Zurich, Switzerland

^cEuropean Centre for Living Technology (ECLT), Ca' Bottacin, Dorsoduro 3911, 30123, Venice, Italy

Article history:

Speaker verification, speaker clustering, dynamic features, prosodic features, deep learning, explainable AI (XAI)

ABSTRACT

While deep neural networks have shown impressive results in automatic speaker recognition and related tasks, it is dissatisfactory how little is understood about what exactly is responsible for these results. Part of the success has been attributed in prior work to their capability to model supra-segmental temporal information (SST), i.e., learn rhythmic-prosodic characteristics of speech in addition to spectral features. In this paper, we (i) present and apply a novel test to quantify to what extent the performance of state-of-the-art neural networks for speaker recognition can be explained by modeling SST; and (ii) present several means to force respective nets to focus more on SST and evaluate their merits. We find that a variety of CNN- and RNN-based neural network architectures for speaker recognition do not model SST to any sufficient degree, even when forced. The results provide a highly relevant basis for impactful future research into better exploitation of the full speech signal and give insights into the inner workings of such networks, enhancing explainability of deep learning for speech technologies.

© 2024 Elsevier Ltd. All rights reserved.

1. Introduction

Deep neural networks (DNNs) have become extremely effective in speaker recognition (SR) and its sub-tasks like speaker verification (SV), identification (SI) or clustering (SC) [1]. Despite this success, deep learning remains driven by empiricism [2], and the available theoretical insights into its workings [3] all the more underline what is yet not understood about how and why DNNs arrive at such a high performance [4], leaving much room for improved explainability of such models [5] also to guide future research.

Meanwhile, the key to human top performance in SR (specifically, in challenging environments) is to make use of a comprehensive variety of spectro-temporal acoustic-phonetic information in speech [6, 7]. Particularly, short-term spectral information, equating to frame-based acoustic information (**FBA**)

in automatic systems, is supplemented in humans by supra-segmental temporal information (**SST**), also referred to as speech prosody. SST varies between individuals [8, 9] and is beneficial for automatic SR systems [10]. Latter authors provided first evidence that succeeding in adequately modeling SST in addition to FBA holds the potential for an order of magnitude less SR errors. Further evidence has been provided using convolutional and recurrent deep learning architectures [11–13], claiming that the achieved gains are due to the superior sequence modeling capabilities of the DNNs that are “successfully capturing prosodic information” [13] – without explicitly testing this. In a similar vein [14, 15]: Zhao et al. [14] argue that their BLSTM-enhanced [16] SV DNN is “*supposedly* phonetically aware” and modeling “context information, which *could* facilitate the ResNet to [...] suppress the environmental variations”, *because* BLSTM layers have the capability to model long ranges. Yet, these claims have never been verified.

If DNNs would not model SST adequately (but achieve their superior results otherwise by focusing on FBA alone), this would imply that the predicted performance gains [10] are still

^{**}Corresponding author
e-mail: neud@zhaw.ch (Daniel Neururer), volker.dellwo@uzh.ch (Volker Dellwo), stdm@zhaw.ch (Thilo Stadelmann)

²D. Neururer and T. Stadelmann contributed equally to this work.

to be realized. More specifically, if it could be quantified to what extent state-of-the-art DNN-based SR systems actually do or do not exploit SST, this would (a) add explanation to a high-performing but opaque class of SR models, (b) show specific directions for targeted future research (concretely targeting the modelling of SST, if it turns out to be under-exploited), and (c) verify theoretical as well as empirical findings in earlier studies, practically guiding future developments. This way, the found discrepancies between DNNs theoretical capabilities and their practical workings are not just uncovered, but can be reconciled in the future.

In this paper, we experimentally analyze this hypothesis of superior modeling of supra-segmental temporal features through contemporary DNNs made in prior work, suspecting that it happens less than assumed. Our motivation is to better understand and improve DNNs in their ability to model fundamental aspects of the speech signal inherently. Hence, we present a novel approach to quantify what amount of sequence modeling is actually occurring, and find the hypothesis of superior SST modeling has to be rejected: Current DNNs lazily rest on phone-level acoustic features alone (cf. Sec. 2). We offer further analyses of this undesirable “*deep cheating*” phenomenon (cf. Sec. 3) by gradually removing the speaker-discriminant information contained in FBA features through innovative experimental setups, thus forcing models to rely on other sources of information. Still they do not switch to model available SST, as evidenced in extensive experiments using diverse state-of-the-art DNN architectures for SV using TIMIT [17] and VoxCeleb [18] benchmarks.

To the best of our knowledge, this paper represents the first systematic study into the modeling of SST by DNNs. Its contributions stand further out in several ways: First, our results explain what DNNs for SR do and do not model, namely that they overfit on the easily exploitable FBA to the point where they nearly ignore SST information, rectifying earlier published conjectures. Second, this opens strategic directions for future SR research that are perpendicular to other current research trends, namely to inquire into better exploitation of SST (cf. Sec. 4) to realize the predicted performance gain of one order of magnitude less errors [10]. Third, the developed benchmark suite of experimental protocol, test metric and established results makes progress in this direction quantifiable.

2. Time scrambling approach to quantify SST exploitation

2.1. Objective and related work

We study SR DNNs that receive spectrogram-like input and are interested in quantifying to which extent they rely on FBA, i.e., spectral characteristics as contained in a single frame of MFCCs [19], and to which extent they exploit SST, i.e., information that is contained in the trajectory over many frames, like intonation and rhythm. Works like [13, 14] assert that such sequence learning is happening automatically because it is a reasonable explanation for the achieved superior results, given the general sequence learning capabilities of the models. On the other hand, Soleymani et al. [20] do not rely on CNNs to pick

up SST automatically just via filters that extend in time: In addition to feeding mel-spectrograms into convolutional layers to extract FBA, they feed hand-crafted prosodic features for late fusion into the DNN to achieve prosody-enhanced SV.

The survey by Bai and Zhang [21] shows that these two views are omnipresent in the literature: Some works use DNN architectures inspired by computer vision [22, 23], thereby asserting that exploiting dynamics will happen via filters and average pooling along the time axis in the same way that such architectures pick up image information along the x-axis automatically. According to Bai and Zhang, this is “the most common [temporal] pooling function”. At the same time, they survey works that deal explicitly with integrating FBA over time, either via collecting statistics [24, 25], applying self-attention [26] or performing a trainable soft clustering of FBA [27]. In this section, we lay the foundation for an informed decision on these options by quantifying how well DNNs *inherently* capture SST in spectrogram-like input without additional explicit care for temporal dynamics.

2.2. Methodology and experimental setup

The authors of [10] used a simple test to probe human reliance on SST: They randomly shuffled the columns of a spectrogram, re-synthesis the result back to audio, and conducted human SR experiments. This way, they left all FBA information intact but removed all SST within said spectrograms. Inspired by this time scrambling approach, we aim to quantify the amount of actual SST exploitation in a DNN by comparing SR performance on original input data with performance on time-scrambled input that contains no SST: Starting from a mel-spectrogram input (front-end processing similar to [13] for comparability), this is achieved by randomizing the order of speech frames, i.e., columns in the spectrogram, thereby removing all original sequence information. The anticipated drop in recognition performance would confirm an anti-proportionally good exploitation of SST through DNNs, while no drop would at least mean that the DNN can compensate missing SST equivalently with the remaining FBA information, hence does not rely on SST exploitation.

Specifically, we consider three ways to extract fixed-length time-scrambled *segments* from a variable-length *utterance* (cf. Fig. 1) according to the following reasoning: **OS** segments will contain FBA and SST, offering a DNN all options to learn these lower- and higher-level features. **SS** segments will, due to a random trajectory of frames, contain no speaker-specific dynamics, offering DNNs only the option to learn about FBA. **SU** segments constitute certain middle ground, biased towards FBA: While still containing no SST, they draw frames from the full utterance rather than a limited segment, offering a richer sample of FBA.

We carried out the following initial experiments on the TIMIT database which provides a controlled acoustic recording environment and normalized utterances (630 speakers under clean studio conditions, 10 sentences per speaker of on average 3s length, 462 speakers in the training set). This laboratory setup enabled us to first study voice recognition performance of a DNN model *per se*, without complications induced through

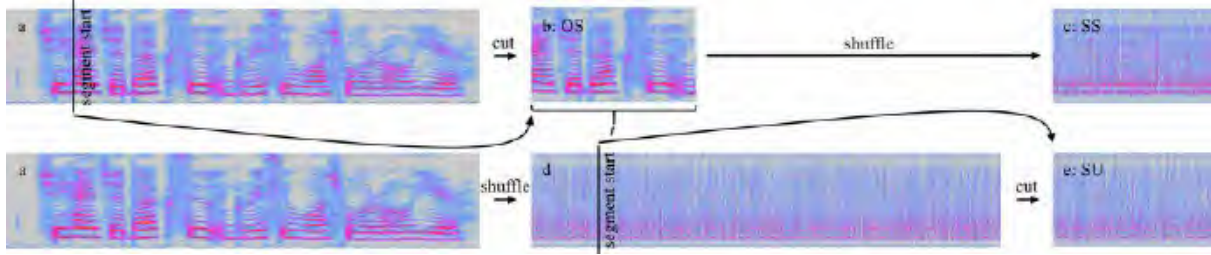


Fig. 1. Segment creation with and without SST: Starting (a) from a spectrogram per varying-length utterance, we extract segments of fixed length t from a starting point according to 3 segment-drawing strategies as follows. Original Segment (OS): just cut out (b) the respective part; Shuffled within Segment (SS): additionally, shuffle (c) the columns of the previous output; Shuffled within Utterance (SU): globally shuffle all columns (d) prior to cutting (e).

the environment (background noise, cross talk, etc.). As our main interest is in understanding and improving DNN modeling for SR (in contrast to improving a complete SR pipeline), we used three different DNN architectures to sample the space of simple to advanced models without explicit care for temporal modeling: (a) As a simple baseline, an adapted version of the vanilla convolutional network (CNN) of [12], sped up by a Cos-Face loss function [28] instead of the computationally heavy KL divergence; (b) as the first model that claimed to model prosody, the recurrent neural network (RNN) of [13]; (c) as a recent architecture, the ResNet34s (ResNet) of [27] that innately contains a GhostVLAD layer to aggregate FBA per segment, adapted to the experimental setup of the previous models with respect to front-end processing (see below); (d) additionally, to account for recent developments in state-of-the-art approaches, the Fast ResNet-34 (F-ResNet) of [29]³ that also serves as a competitive baseline in the VoxSRC SR benchmarking efforts [1], used here with its original parameters also for front-end processing (and hence not directly comparable to the first three models). Utmost care has been taken to keep all not explicitly mentioned parameters (here or below) for, e.g., front-end processing, DNN architecture and training, embedding extraction, and classification equal to the respective original work in order to allow for comparisons. Hence, the first three models allow a comparison with prior work on modeling dynamic voice features with DNNs, while the fourth model allows a comparison with the current state-of-the-art. We expected all models to suffer performance loss in increasing magnitude when fed with shuffled frames within segments due to increasing capability to model SST.

For training of the CNN, RNN, and ResNet model, to warrant backward comparability with prior work, in each of 128 epochs we draw one $t = 1$ s long segment (to account for short utterances [30]) from a random location per utterance in the training set, using a batch size of 100 and otherwise a similar experimental setup as [13]. For the F-ResNet, to keep forward comparability with current SR benchmarking efforts, we keep all hyperparameters similar to the experimental setup in [29], specifically using $t = 2$ s long training segments ($t = 4$ s for evaluation), 500 epochs and a batch size of 800. We repeat this for each of the OS/SS/SU strategies to draw segments with/without

Table 1. SC results on TIMIT [MR μ/σ]. Bold font indicates best results per model, cell coloring scales with quality per model. Cells marked by $\downarrow$ are discussed in the text.

$\downarrow$ training / test $\rightarrow$		OS	SU	SS
CNN [12]	OS	a$\downarrow$ 0.00 σ0.00	9.75 σ 0.94	a $\downarrow$ 9.00 σ 2.15
	SU	8.50 σ 2.42	e$\downarrow$ 0.50 σ0.61	1.75 σ 0.61
	SS	a $\downarrow$ 9.00 σ 1.66	1.00 σ 0.50	b $\downarrow$ 1.25 σ 0.00
RNN [13]	OS	d $\downarrow$ 1.25 σ 1.12	2.75 σ 0.94	f $\downarrow$ 2.75 σ 0.50
	SU	3.75 σ 1.37	0.00 σ0.00	2.50 σ 1.58
	SS	2.00 σ 1.00	1.25 σ 0.79	d $\downarrow$ 0.25 σ 0.50
ResNet [27]	OS	d $\downarrow$ 1.00 σ0.94	8.25 σ 4.78	f $\downarrow$ 11.50 σ 4.29
	SU	2.50 σ 1.77	1.00 σ0.50	3.00 σ 1.27
	SS	2.75 σ 0.94	1.25 σ 1.12	d $\downarrow$ 1.00 σ0.94
F-ResNet [29]	OS	11.50 σ 2.15	37.50 σ4.18	33.75 σ 4.18
	SU	16.50 σ 2.42	5.75 σ 1.70	4.25 σ 1.00
	SS	15.50 σ 2.57	6.75 σ 1.50	j $\downarrow$ 3.75 σ0.79

Table 2. SV results on TIMIT [EER μ/σ]. As with the SC results on TIMIT in Tab. 1, the F-ResNet is out of competition here due to different front-end processing.

$\downarrow$ training / test $\rightarrow$		OS	SU	SS
CNN [12]	OS	c $\downarrow$ 6.38 σ 0.12	12.02 σ 0.51	f $\downarrow$ 11.90 σ 0.46
	SU	8.55 σ 0.49	g $\downarrow$ 5.55 σ 0.06	6.12 σ 0.12
	SS	8.16 σ 0.42	5.33 σ0.18	c $\downarrow$ 5.78 σ 0.16
RNN [13]	OS	d $\downarrow$ 3.53 σ0.07	4.19 σ 0.09	f $\downarrow$ 3.90 σ 0.12
	SU	3.99 σ 0.16	3.78 σ 0.10	3.66 σ 0.13
	SS	4.00 σ 0.07	3.89 σ 0.06	d,g $\downarrow$ 3.54 σ 0.03
ResNet [27]	OS	4.96 σ0.19	10.34 σ 1.56	f $\downarrow$ 9.21 σ 1.15
	SU	6.59 σ 0.25	6.25 σ 0.23	6.37 σ 0.35
	SS	5.89 σ 0.25	6.11 σ 0.31	g $\downarrow$ 5.80 σ 0.11
F-ResNet [29]	OS	12.20 σ 0.25	23.41 σ1.73	20.47 σ 1.50
	SU	15.12 σ 0.84	10.46 σ 0.28	9.69 σ 0.20
	SS	15.91 σ 0.90	9.86 σ 0.11	j $\downarrow$ 8.95 σ0.13

SST, thus training three separate versions of each model. We use these models to extract embeddings for two downstream tasks: For SV, we pair every sentence in the TIMIT test set with every other sentence therein, and evaluate the equal error rate (EER) as the standard metric used for SV [1], resulting in 2.82 mio. pairwise comparisons. For SC, we perform hierarchical clustering of 2 utterances (comprising 2 and 8 concatenated sentences) per 40 speakers as in [13], measured by the mis-

³Sourced from https://github.com/clovaai/voxceleb_trainer.

classification rate (MR) as the established measure [11] (6,400 pairwise comparisons). For each reported EER/MR, we average over 5 train/test runs and report mean and SD.

2.3. Results and discussion

Results are shown in Tabs. 1 and 2. For the analysis, we first focus on SC results (Tab. 1) and original timing vs. segment-wise shuffled (rows/columns marked with OS/SS), ignoring the rest. With reference to the lowercase letters in the cells marked by $\Downarrow$, we can say: (a) Looking at the CNN model only, everything appears as expected as best results ($MR = 0$ in all 5 runs, the best result ever reported on this benchmark) are achieved for training and testing with OS, and worst results are achieved when SS is used for training *or* evaluation. (b) But already looking at training *and* testing using SS raises doubts w.r.t. actual SST exploitation: $MR = 1.25$ is a very good result for the task, still outperforming the previous state of the art [13], but not using any SST (as it has been removed from the data). (c) Taking, secondly, SV results (Tab. 2) into account, these doubts are confirmed. For the CNN model, training/testing using SS/SS outperforms OS/OS by 0.6 difference in EER. (d) A similar picture is seen for the RNN on both tasks and the ResNet on SC, where SS/SS either outperforms or (within σ) equals OS/OS. (e) When adding SU for training and/or testing into the picture, this tendency is confirmed: SU/SU outperforms SS/SS and is almost on par with OS/OS (cf. a). (f) However, models do pick up something about inter-frame relationships as in cross conditions like OS training and SS testing, a performance drop is evident for the CNN and ResNet (not so for the RNN). We argue that this is rather the effect of mismatched train/test conditions as is usual in any machine learned model; previous discussion (specifically, that best results are achieved using SU in several cases) has shown that (almost) nothing useful is extracted from the trajectory. (j) The F-ResNet, which uses the front-end processing and architectural parameters optimized for SV on the by orders of magnitude larger VoxCeleb dataset, has expected difficulties with the tiny TIMIT benchmark (which could be cured by proper pre-training, but would not help the purpose of these experiments). It hence ran out of competition here, but will play a major role in later experiments (see Section 3). For now, it suffices to say that it behaved quite similar to the ResNet. That the best scores are achieved using the SS/SS setup on both tasks can be explained with the general lack of sufficient amounts of training data, specifically when using longer segments, and that in this regime SS provides most FBA information per segment (i.e., the higher sample efficiency can partially make up for lacking training data).

Summarizing, we found that completely randomizing the order of speech frames in segments used for training and evaluation still produced state-of-the-art or even better SV (e.g., SU/SU for the RNN) and SC (e.g., SS/SS for the RNN) results. This is notwithstanding the obviously also state-of-the-art results of the OS/OS setup for several models (e.g., the CNN for SC; ResNet for SV) – the point here is not that the results using SS or SU are in any way superior to OS, but that they are sometimes not worse. Given this analysis, we believe to have strong evidence that the tested DNN models do not rely on SST (they



Fig. 2. Visualization of FBA equalization: Compressed spectrograms of the same sentence (SA1) of a male (a: MDAB0) and female (b: FCJF0) TIMIT speaker with derived synthesized (a1/b1) and noise-vocoded (a2/b2) variants.

might model them, but are ultimately able to produce similar results without them). We conjecture that this is due to purely modeling FBA is easy and efficient enough: It quickly brings the loss down to a local minimum from which the training cannot recover as switching features to a different set would introduce too high intermediate losses again. This is in accordance with studies claiming that DNNs are lazy in learning complex concepts when easy ones work already [31], and tend to learn shortcuts [32]. To put it drastically, DNNs “cheat” by ignoring the harder task of modeling dynamics if they can.

Technically speaking, the model overfits to FBA information to the point of neglecting other evidence. An ablation study gives further evidence of this: Remarkably, replacing the CNN in the TIMIT SV task with a simplistic model per speaker that only holds a row-wise average of all the speaker’s training spectrograms (i.e., a single-Gaussian model with unit variance) outperforms the CNN. Apparently, having access to the full frequency resolution of the spectrogram at decision time is a larger advantage to the simple model than pooling in frequency *and* time is for the CNN. Overcoming this problem calls for a novel form of regularization on the task-level, which will be explored in the following section.

3. Regularization approaches to increase SST exploitation

3.1. Objective and context

Our objective here is to find ways to force models to revert to SST. As one implication of the results of Sec. 2 is that the predicted performance improvements [10] from exploiting SST are still to be harvested, this holds large prospects. It could be realized, e.g., by combining a model that focuses on FBA with one regularized to focus on SST [33].

3.2. Methodology and experimental setup

First, we test the conjecture that DNNs only model those features necessary to solve the task, starting with the easiest to learn. Therefor, we increase the difficulty of the task by including an acoustically more challenging dataset and check for

Table 3. SV results on VoxCeleb (left) and noise-vocoded (middle) as well as resynthesized (right) TIMIT [EER μ/σ of 5 runs].

		VoxCeleb			TIMIT-NV			TIMIT-Syn		
↓ training / test →		OS	SU	SS	OS	SU	SS	OS	SU	SS
CNN [12]	OS	gll 25.75 σ 0.13	37.23 σ 0.74	36.96 σ 0.78	32.56 σ 0.62	35.32 σ 0.46	35.41 σ 0.55	46.24 σ 0.18	48.94 σ 0.15	48.97 σ 0.23
	SU	32.70 σ 0.34	27.04 σ 0.34	27.99 σ 0.30	35.16 σ 0.52	hll 30.39 σ 0.30	30.91 σ 0.47	47.26 σ 0.15	45.98 σ 0.34	46.16 σ 0.27
	SS	33.26 σ 0.29	27.91 σ 0.32	28.50 σ 0.28	35.25 σ 0.69	30.63 σ 0.38	31.23 σ 0.27	47.14 σ 0.22	45.88 σ 0.12	ijl 45.66 σ 0.12
RNN [13]	OS	gll 20.67 σ 0.23	30.67 σ 0.36	30.00 σ 0.32	hll 19.34 σ 0.16	27.20 σ 0.42	26.12 σ 0.44	ijl 40.39 σ 0.07	44.29 σ 0.65	42.43 σ 1.40
	SU	26.20 σ 0.18	22.02 σ 0.10	23.57 σ 0.09	22.95 σ 0.24	21.48 σ 0.40	21.15 σ 0.25	43.63 σ 0.35	41.93 σ 0.26	41.64 σ 0.25
	SS	28.28 σ 1.30	26.30 σ 0.59	26.58 σ 0.84	22.82 σ 0.40	21.89 σ 0.25	21.04 σ 0.12	43.62 σ 0.21	42.55 σ 0.34	41.53 σ 0.23
ResNet [27]	OS	gll 12.49 σ 0.15	34.11 σ 0.54	32.19 σ 0.39	hll 21.12 σ 0.43	37.83 σ 1.17	36.57 σ 1.45	ijl 40.33 σ 1.32	47.28 σ 2.06	46.60 σ 2.02
	SU	22.05 σ 0.43	19.08 σ 0.26	20.02 σ 0.16	27.03 σ 0.63	23.38 σ 0.41	24.02 σ 0.25	43.44 σ 0.86	42.97 σ 0.51	42.65 σ 0.59
	SS	20.74 σ 0.46	21.02 σ 0.34	20.36 σ 0.23	27.25 σ 1.37	23.57 σ 0.46	23.32 σ 0.58	42.48 σ 0.45	43.07 σ 0.72	41.59 σ 0.36
F-ResNet [29]	OS	gll 2.39 σ 0.05	25.02 σ 1.21	23.45 σ 1.29	kll 24.72 σ 0.54	37.17 σ 0.57	34.54 σ 0.66	kll 39.06 σ 0.50	47.52 σ 0.58	46.48 σ 1.02
	SU	11.00 σ 0.57	6.69 σ 0.11	6.58 σ 0.15	31.46 σ 0.86	22.40 σ 0.42	21.67 σ 0.49	42.85 σ 0.47	40.10 σ 0.11	40.19 σ 0.10
	SS	10.85 σ 0.43	7.02 σ 0.24	6.60 σ 0.19	31.44 σ 1.11	22.04 σ 0.12	kll 21.24 σ 0.17	43.23 σ 0.29	40.44 σ 0.18	40.36 σ 0.20

an increase of actual SST modeling via the test established before, using the VoxCeleb corpus while keeping other parameters equal to the TIMIT analysis in Sec. 2 (i.e., for the CNN, RNN, and ResNet models we keep the experimental setup compatible with [13] while for the F-ResNet model we keep it compatible with [29]). VoxCeleb1 contains 148,642 utterances of varying length (few seconds up to several minutes) from 1,211 different speakers recorded in the wild, thus holding a variety of background noises including cross talk; VoxCeleb2 contains more than 1 million utterances from 5,994 speakers. We use the standard experimental protocol of Chung et al. to train a SV system on VoxCeleb2, and evaluate SV performance on the “hard” test set of VoxCeleb1 [29]. We omit SC results on VoxCeleb as experiments in Sec. 2 led to similar conclusions for both tasks.

Second, we aim at actively nudging models to learn SST by making FBA less attractive, inspired by the previous approach that scrambled SST. While scrambling the frequency axis by naïvely randomizing the rows in a spectrogram would effectively remove SST as well (as it depends on the evolution of sound over frequency band borders), we instead propose the regularization strategy of largely *equalizing* (instead of randomizing) FBA amongst speakers while retaining their discriminative SST. Note that this is done specifically to study the effect of devalued FBA on SST-modeling by our DNNs, not as a general way to augment speech data [34]. Specifically, we chose two equalization strategies (cf. Fig. 2) on the TIMIT corpus: (a) Noise-vocoding the sentences with just 4 broad frequency bands [35], modulated by the original speakers’ energy levels, to *reduce* spectral differences; and (b) using a custom version of the Slang TTS [36] speech synthesizer (able to use speaker-specific phoneme-level timing annotations and energy contours from the TIMIT corpus as additional input) to recreate every sentence from its text transcript with an identical synthetic voice, thus *eliminating* spectral differences but keeping some SST.

3.3. Results and discussion

SV results on VoxCeleb are shown in the left part of Tab. 3. Comparing them with SV on standard TIMIT (Tab. 2) and using

the cell annotations marked by $\downarrow$, the following is noteworthy: (g) On TIMIT, competitive results are achieved using SU/SU (CNN) and SS/SS (RNN, ResNet) that are only marginally worse than the best result per model and in particular strong in comparison with OS/OS. On VoxCeleb, OS/OS now clearly is better than all other combinations that include randomization by a larger margin. While the absolute scores are bad for the two simpler models, the ResNet shows a reasonable EER using OS/OS on VoxCeleb, only about 3 times worse than the best result on the much simpler TIMIT and about twice as good as the best result involving any randomization. The F-ResNet shows state-of-the-art performance under OS/OS (i.e., normal) conditions, thereby conforming that the experimental setup and codebase used for these experiments is sound. We conclude that providing a more challenging task (one that cannot be solved relying on simpler frequency-domain features alone, as research prior to the deep learning era has shown [37]) stimulates the exploitation of SST in models to some degree, depending on the ability of the model – as all conditions with scrambled SST fall far behind in performance (by a factor of ≥ 2.75 for the best model, F-ResNet).

The middle part of Tab. 3 contains the results on noise-vocoded TIMIT, where speakers’ individual timbre has been largely removed. Again comparing with the respective results on standard TIMIT, it is noteworthy that (h) best results are now achieved by OS/OS for the RNN and ResNet. However, the CNN still has best results involving random timing, and also for the RNN and ResNet models, the margin for OS/OS is small and EERs are 4-5 times higher than on standard TIMIT. We conclude that the effect seen on VoxCeleb (a harder task makes the models start learning SST) is visible to some degree, but less pronounced. (i) The same is true for the results on resynthesized TIMIT (cf. right part of Tab. 3), except that best results for the CNN are achieved using another form of randomization and the EERs are 8-11 times worse here. (k) It is different for the F-ResNet, where best results on noise-vocoded TIMIT are still achieved using SS/SS (and only with a very thin margin using OS/OS on resynthesized TIMIT). We conjecture that this is again due to the tininess of the TIMIT database for training this large model (see (j) discussed in Sec. 2).

We conclude that diminishing the dominance (i.e., speaker discriminativeness) of FBA partially (using noise vocoding) or fully (through resynthesis) brings forth some exploitation of SST for SV, but not optimally (which would be evidenced not necessarily by lower absolute EERs, but by larger margins between top OS/OS results and everything else). Moreover, such an effect is even less pronounced when switching tasks from SV to SC (tables omitted): No benefit of OS/OS can be observed on TIMIT-NV and only little evidence for it is seen on TIMIT-Syn. We conjecture that the attempt to model SST apart from FBE is suboptimal (in accordance with the literature [6, 7] that categorizes SST as of subordinate importance but helpful in addition to FBA). Evidence for this is the experiment on VoxCeleb that shows that under challenging circumstances ResNets can achieve unparalleled results by exploiting FBA and SST jointly.

4. Conclusions, future work and limitations

In this paper, we have presented the first systematic study on learning supra-segmental temporal features by DNNs for SR. Not focusing on presenting a new kind of model or SR methodology, we have instead shown that state-of-the-art CNN, RNN and ResNet models for SV and SC, when trained on clean data, simply ignore any useful supra-segmental temporal cues in the audio signal despite contrary conjectures in the literature and the models' principal abilities to learn such features. We have called this phenomenon "deep cheating". It is relevant since related work provided evidence [10] that improved modeling of such higher-level features should result in one order of magnitude lower error rates in related tasks and hence holds a key for targeted future research, guided by our test to quantify actual SST exploitation. It is also of importance in the context of explaining *how* DNNs achieve their superior results (XAI), where our explanation goes beyond activation visualization to explain individual classifications towards a broader understanding of signal processing by DNNs.

Furthermore, we have presented two approaches to force DNNs to exploit SST, and measured their effectiveness: (a) Increasing task difficulty by using acoustically more challenging data (VoxCeleb instead of TIMIT), and (b) removing the discriminative power of FBA by equalizing speakers' timbre. The results indicate that both approaches achieve respective results nominally, thereby confirming other studies that attest DNNs laziness in modeling only the easiest available features to solve a given task. Theoretical and empirical studies suggest that scaling up training time might help overcome such imperfect local minima [38].

We have conducted extensive experiments to verify the correctness and stability of our results for a wide range of design choices. Our claims hold for reasonable settings of the hyperparameters learning rate, number of epochs, segment and hop length, and embedding size; when using the original loss functions of published models instead of speed-ups; with varying number of frequency bands for the noise vocoder or using MBROLA [39] as another synthesizer; and for evaluating on VoxCeleb2. TIMIT, though small, is a sound basis for our findings (cf. [38]): It has been used successfully for this purpose by

the community before; we do not observe problems with overfitting for all but the F-ResNet model, although by construction of the mini batches, we only exploit a fraction of the available training data; it contains pure voices without exogenous difficulties (noise, brevity, ...), offering to study SR capability in isolation and hence granting an unbiased look at DNNs' abilities for voice modeling.

However, we have also shown that attempting to learn SST apart from FBA results in severely underperforming models that verge on random SV results and are even less helpful for SC. Hence, our results are preliminary with respect to finding better ways of exploiting speaker-specific SST with DNNs. Thinking of perceivable instantiations of SST like personal linguistic melody, it is elusive how such strong rhythmic-prosodic patterns are not picked up by any of the most capable general pattern recognition methods we know today, deep neural networks. Future work should therefore concentrate on finding inductive biases for deep networks that fully exploit the time axis for speaker specificity. Inspired by how auto-regressive self-supervised learning in text processing works [40] and building on the success of respective large language models, transformer-based architectures and large-scale pre-training could be a way to integrate handling of dynamic features either directly into the model or into methods for speech augmentation. Such work should use the test presented in Sec. 2 to benchmark the success of actually exploiting STT. Our code is available online at <https://tinyurl.com/deepcheating>.

Acknowledgments

The authors thank Bernd Freisleben, Claude Olivier Lehmann, Frank-Peter Schilling and Richard Hahnloser for encouraging discussions, and their collaborators on voice recognition methodology over the years: Oliver Dürr, Yanick X. Lukic, Carlo Vogt, Feliks Hibraj, Sebastiano Vascon, Marcello Pelillo, Patrick Gerber, Sebastian Glinski-Haefeli, Benjamin Neusser, Savin Niederer, Timon Gyax, Jörg Egli, Niclas Simmler, Amin Trabi, Jan Sonderegger, Patrick Walter, Christian Lauener, Pascal Sager, and Sydney M. Nguyen.

References

- [1] A. Brown, J. Huh, J. S. Chung, A. Nagrani, D. Garcia-Romero, A. Zisserman, VoxSRC 2021: The third VoxCeleb speaker recognition challenge (2022). arXiv:2201.04583.
- [2] L. Tugener, J. Schmidhuber, T. Stadelmann, Is it enough to optimize CNN architectures on ImageNet?, *Frontiers in Computer Science* (2022).
- [3] H. W. Lin, M. Tegmark, D. Rolnick, Why does deep and cheap learning work so well?, *Journal of Statistical Physics* 168 (6) (2017) 1223–1247.
- [4] T. J. Sejnowski, The unreasonable effectiveness of deep learning in artificial intelligence, *PNAS* 117 (48) (2020) 30033–30038.
- [5] M. Ivanovs, R. Kadikis, K. Ozols, Perturbation-based methods for explaining deep neural networks: A survey, *Pattern Recognit. Lett.* 150 (2021) 228–234.
- [6] P. Rose, *Forensic speaker identification*, Taylor & Francis, London and New York, 2002.
- [7] J. H. Hansen, T. Hasan, Speaker recognition by machines and humans: A tutorial review, *IEEE Signal processing magazine* 32 (6) (2015) 74–99.
- [8] A. Leemann, M.-J. Kolly, V. Dellwo, Speaker-individuality in suprasegmental temporal features: Implications for forensic voice comparison, *Forensic science international* 238 (2014) 59–67.

- [9] S. Shah Nawazuddin, N. Adiga, H. K. Kathania, B. T. Sai, Creating speaker independent ASR system through prosody modification based data augmentation, *Pattern Recognit. Lett.* 131 (2020) 213–218.
- [10] T. Stadelmann, B. Freisleben, Unfolding speaker clustering potential: a biomimetic approach, in: *Proc. ACM MM*, 2009, pp. 185–194.
- [11] Y. X. Lukic, C. Vogt, O. Dürr, T. Stadelmann, Speaker identification and clustering using convolutional neural networks, in: *Proc. MLSP*, 2016, pp. 1–6.
- [12] Y. X. Lukic, C. Vogt, O. Dürr, T. Stadelmann, Learning embeddings for speaker clustering based on voice equality, in: *Proc. MLSP*, 2017, pp. 1–6.
- [13] T. Stadelmann, S. Glinski-Haefeli, P. Gerber, O. Dürr, Capturing suprasegmental features of a voice with rnns for improved speaker clustering, in: *Proc. ANNPR*, 2018, pp. 333–345.
- [14] Y. Zhao, T. Zhou, Z. Chen, J. Wu, Improving deep CNN networks with long temporal context for text-independent speaker verification, in: *Proc. ICASSP*, 2020, pp. 6834–6838.
- [15] F. Ye, J. Yang, A deep neural network model for speaker identification, *Applied Sciences* 11 (8) (2021) 3603.
- [16] A. Graves, A.-R. Mohamed, G. Hinton, Speech recognition with deep recurrent neural networks, in: *Proc. ICASSP*, 2013, pp. 6645–6649.
- [17] W. M. Fisher, The DARPA speech recognition research database: specifications and status, in: *Proc. DARPA Workshop on Speech Recognition*, 1986, pp. 93–99.
- [18] A. Nagrani, J. S. Chung, A. Zisserman, VoxCeleb: a large-scale speaker identification dataset, in: *Proc. Interspeech*, 2017, pp. 2616–2620.
- [19] S. B. Davis, P. Mermelstein, Evaluation of acoustic parameters for monosyllabic word identification, *J. Acoust. Soc. Am.* 64 (S1) (1978) S180–S181.
- [20] S. Soleymani, A. Dabouei, S. M. Iranmanesh, H. Kazemi, J. Dawson, N. M. Nasrabadi, Prosodic-enhanced siamese convolutional neural networks for cross-device text-independent speaker verification, in: *Proc. BTAS*, 2018, pp. 1–7.
- [21] Z. Bai, X.-L. Zhang, Speaker recognition based on deep learning: An overview, *Neural Netw.* 140 (2021) 65–99. doi:<https://doi.org/10.1016/j.neunet.2021.03.004>.
- [22] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: *Proc. CVPR*, 2016, pp. 770–778.
- [23] T. Stadelmann, V. Tolkachev, B. Sick, J. Stampfli, O. Dürr, Beyond imagenet: deep learning in industrial practice, in: *Applied Data Science*, Springer, 2019, pp. 205–232.
- [24] D. Snyder, D. Garcia-Romero, D. Povey, S. Khudanpur, Deep neural network embeddings for text-independent speaker verification, in: *Proc. Interspeech*, 2017, pp. 999–1003.
- [25] H. N. Pinheiro, T. I. Ren, A. G. Adami, G. D. Cavalcanti, Variational dnn embeddings for text-independent speaker verification, *Pattern Recognit. Lett.* 148 (2021) 100–106.
- [26] Y. Wu, C. Guo, H. Gao, X. Hou, J. Xu, Vector-based attentive pooling for text-independent speaker verification, in: *Proc. Interspeech*, 2020, pp. 936–940.
- [27] W. Xie, A. Nagrani, J. S. Chung, A. Zisserman, Utterance-level aggregation for speaker recognition in the wild, in: *Proc. ICASSP*, 2019, pp. 5791–5795.
- [28] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, W. Liu, CosFace: Large margin cosine loss for deep face recognition, in: *Proc. CVPR*, 2018, pp. 5265–5274.
- [29] J. S. Chung, J. Huh, S. Mun, M. Lee, H. S. Heo, S. Choe, C. Ham, S. Jung, B.-J. Lee, I. Han, In defence of metric learning for speaker recognition, in: *Proc. Interspeech*, 2020, pp. 2977–2981.
- [30] T. Stadelmann, B. Freisleben, Dimension-decoupled gaussian mixture model for short utterance speaker recognition, in: *Proc. ICPR*, 2010, pp. 1602–1605.
- [31] T. Mansour, Deep neural networks are lazy: on the inductive bias of deep learning, Master's thesis, MIT (2019).
- [32] A. J. DeGrave, J. D. Janizek, S.-I. Lee, AI for radiographic COVID-19 detection selects shortcuts over signal, *Nature Machine Intelligence* 3 (7) (2021) 610–619.
- [33] M. Ganaie, M. Hu, A. Malik, M. Tanveer, P. Suganthan, Ensemble deep learning: A review, *Eng. Appl. Artif. Intell.* (2022).
- [34] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, Q. V. Le, SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition, in: *Proc. Interspeech*, 2019, pp. 2613–2617. doi:[10.21437/Interspeech.2019-2680](https://doi.org/10.21437/Interspeech.2019-2680).
- [35] R. V. Shannon, F.-G. Zeng, V. Kamath, J. Wygonski, M. Ekelid, Speech recognition with primarily temporal cues, *Science* 270 (5234) (1995) 303–304.
- [36] SlowSoft GmbH, Slang TTS speech synthesizer, <https://slowsoft.ch/eng/products.html>, accessed: 2022-02-15 (2021).
- [37] T. Stadelmann, Voice modeling methods for automatic speaker recognition, Ph.D. thesis, Philipps-Universität Marburg (2010).
- [38] A. Power, Y. Burda, H. Edwards, I. Babuschkin, V. Misra, Grokking: Generalization beyond overfitting on small algorithmic datasets, *arXiv preprint arXiv:2201.02177* (2022).
- [39] T. Dutoit, V. Pagel, N. Pierret, F. Bataille, O. Van der Vrecken, The MBROLA project: Towards a set of high quality speech synthesizers free of use for non commercial purposes, in: *Proc. ICSLP*, 1996, pp. 1393–1396.
- [40] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, in: *Proc. ICLR*, 2013.

Originally published in *Geothermics* 120, p. 103012, Elsevier, June 2024, DOI 10.1016/j.geothermics.2024.103012.

Machine Learning for Robust Structural Uncertainty Quantification in Fractured Reservoirs

Dashti, Ali^{1*}; Stadelmann, Thilo²; Kohl, Thomas¹

1. Institute of Applied Geosciences, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany
2. Centre for AI, Technikumstrasse 71, Zurich University of Applied Sciences, 8400 Winterthur, Switzerland

* Corresponding author: Ali.dashti@kit.edu

Abstract

Including uncertainty is essential for accurate decision-making in underground applications. We propose a novel approach to consider structural uncertainty in two enhanced geothermal systems (EGSs) using machine learning (ML) models. The results of numerical simulations show that a small change in the structural model can cause a significant variation in the tracer breakthrough curves (BTCs). To develop a more robust method for including structural uncertainty, we train three different ML models: decision tree regression (DTR), random forest regression (RFR), and gradient boosting regression (GBR). DTR and RFR predict the entire BTC at once, but they are susceptible to overfitting and underfitting. In contrast, GBR predicts each time step of the BTC as a separate target variable, considering the possible correlation between consecutive time steps. This approach is implemented using a chain of regression models. The chain model achieves an acceptable increase in RMSE from train to test data, confirming its ability to capture both the general trend and small-scale heterogeneities of the BTCs. Additionally, using the ML model instead of the numerical solver reduces the computational time by six orders of magnitude. This time efficiency allows us to calculate BTCs for 2'000 different reservoir models, enabling a more comprehensive structural uncertainty quantification for EGS cases. The chain model is particularly promising, as it is robust to overfitting and underfitting and can generate BTCs for a large number of structural models efficiently.

Keywords: machine learning, uncertainty quantification, structural uncertainty, EGS

1 Introduction

Numerical simulations of physical systems described by differential equations are essential in engineering. Advancements in hardware have enabled computing units to solve coupled nonlinear differential equations, encompassing a wide range of phenomena, from weather forecasting (Bauer et al., 2015) to blood circulation in living bodies (Doost et al., 2016). However, these methods are computationally intensive and highly sensitive to specific cases. Besides the huge energy consumption of these computational infrastructures (Benoit et al., 2018), their availability is also limited. Furthermore, parameter tuning, sensitivity analysis (Borgonovo and Plischke, 2016), and uncertainty quantification (Abbaszadeh Shahri et al., 2022; Soize, 2017) demand up to millions of simulations.

Machine learning (ML) methods have gained significant traction across various fields (Brunton and Kutz, 2022; Stadelmann et al., 2019), including geothermal applications (Okoroafor et al., 2022). In this context, data-driven and physics-informed ML (physics-informed neural network, PINN) techniques are of great interest (Carleo et al., 2019; Raissi et al., 2019). PINNs and their diverse descendants are ceaselessly flourishing to replace numerical solvers (Karniadakis et al., 2021; Kharazmi et al., 2019; Knapp et al., 2021; Yu et al., 2022); however, their accuracy and time-efficiency for solving complex problems is still a subject of development (Degen et al., 2023).

One of the challenges in geothermal applications is characterizing fluid flow through complex underground networks. While the geometry of a fracture can define the general direction of flow, the local variation of petrophysical properties impacts the specific pathways (Meakin and Tartakovsky, 2009). The enhanced geothermal system (EGS), as an engineered underground reservoir, strongly relies on high flow rate circulation through the impermeable matrix. To enhance the reservoir's permeability, the cold fracturing fluid is injected to create new fractures or reopen the pre-existing ones (e.g. Kohl and Mège, 2007). Hence, a complex underground fracture/flow pattern can be observed in any EGS example like the model presented by Egert et al. (2020).

Integrating local data coming from wells with field measurements like tracer tests (Cao et al., 2020) can provide insights into the EGS situation. Tracer test campaigns usually yield breakthrough curves (BTCs), which are widely used to extract properties of the porous media and fracture network. However, each measuring method is error-prone resulting in inherent uncertainty (Bond, 2015; Wellmann et al., 2010). Therefore, incorporating structural uncertainties in numerical simulations in EGS settings makes the flow forecast more realistic (Zhou et al., 2022).

This study proposes to replace computationally demanding simulations with speedy ML models to quantify structural uncertainty estimations derived from tracer data in two different EGS settings. By state-of-the-art ML methods like decision tree regression (DTR), random forest regression (RFR), and gradient boosting regression (GBR), multifold BTCs are generated on top of pure time-consuming numerical simulations. We train reliable ML models to map geometric data from the uncertain fractures of the EGS reservoir to the simulated BTC. The position of the varying structural elements is used as the input feature, and the entire BTC is chosen as the target variable. The proposed ML model correlates the entire BTC with input features, rather than using a time window to predict the future.

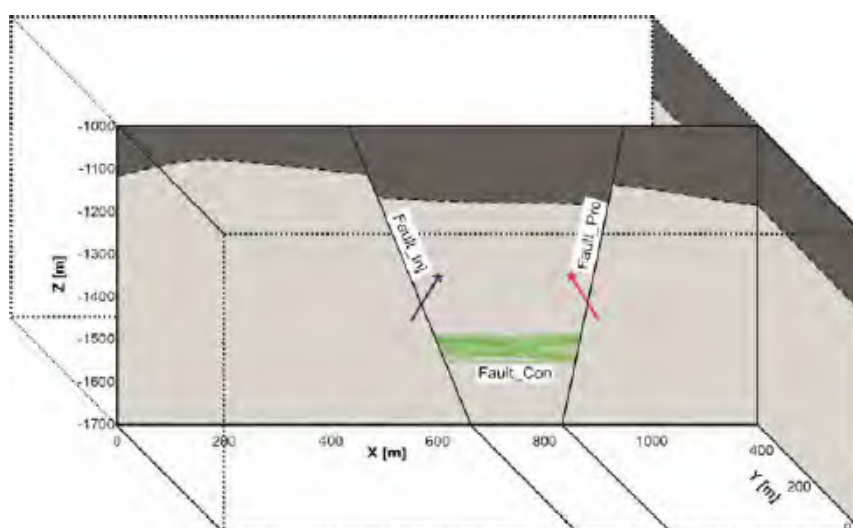
2 Methodology

2.1 Tracer models

Tracer flow in two different cases are applied in this study. The conceptual model introduced by Dashti et al. (2023) is used here as the first case. The model for the first case is called the

74 'simple case' because it is a highly simplified version of an EGS with a doublet configuration.
 75 The conceptual model contains two main transmissive/open faults that are connected to an
 76 injection and production well. There is also an additional sub-horizontal fault/fracture structure
 77 making a connection between the major faults at greater depth. However, data related to this
 78 structure are subject to uncertainty since this fault is located far from the drilling trajectory, and
 79 its existence as a conduit is confirmed only by additional geophysical surveys or hydraulic
 80 testing. Figure 1 provides a schematic view of the model, where two sub-vertical faults intersect
 81 with the injection and production wells and are labeled as Fault_Inj and Fault_Pro, respectively.
 82 The sub-horizontal fault, referred to as Fault_Con in the figure, is represented by thin green
 83 lines, as it connects the two major faults. Dashti et al. (2023) introduced a range of structural
 84 scenarios and perturbed the location of the sub-horizontal fault 50 times to investigate the
 85 impact of structural uncertainty on flow.

86



87

88 Figure 1. A schematic view of the simple case. The two certain sub-vertical faults (Fault_Inj and Fault_Pro) are
 89 shown as continuous black lines and the thinner green lines show traces of the uncertain sub-vertical fault
 90 (Fault_Con). Each green trace makes a unique structural scenario.

91 To comprehensively evaluate the performance of ML methods, a second, more intricate
 92 fracture network model (named as complex case) was developed (Figure 2). The 'complex
 93 case' incorporates seven fractures, with two designated as uncertain. The impact of varying
 94 these two fractures' depth and dip angle on tracer flow was assessed through 100 scenarios.
 95 All scenarios shared identical material properties, while the uncertain fractures' dip and depth
 96 were varied. The modelling assumptions of the complex case are similar to the simple case
 97 which is already addressed in Dashti et al. (2023).

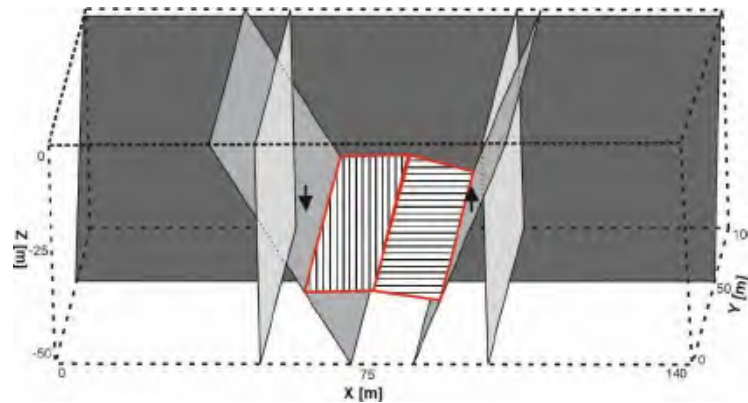


Figure 2. A complex EGS setting with seven fractures. Certain (five) fractures are shown as grey surfaces with varying shades and solid black borders while the two uncertain fractures are highlighted via the thick red border and hashed infill. Two arrows show the location of the injection and production wells.

2.2 Machine learning model

The ML model in this study predicts the tracer concentrations over time, i.e. the BTCs for two cases. Time series estimation for different applications is a well-documented topic (Gudmundsdottir and Horne, 2020; Weigend and Gershenfeld, 1994). For example, Alakeely and Horne (2020) introduced a recurrent neural network to predict the future by incorporating historical data. Such methods predict the system's long-term performance based on a moderate duration of the monitoring data. However, our study predicts the entire time series making the ML models applicable for cases without any historical data.

Due to the nature of the problem, two different strategies are developed.

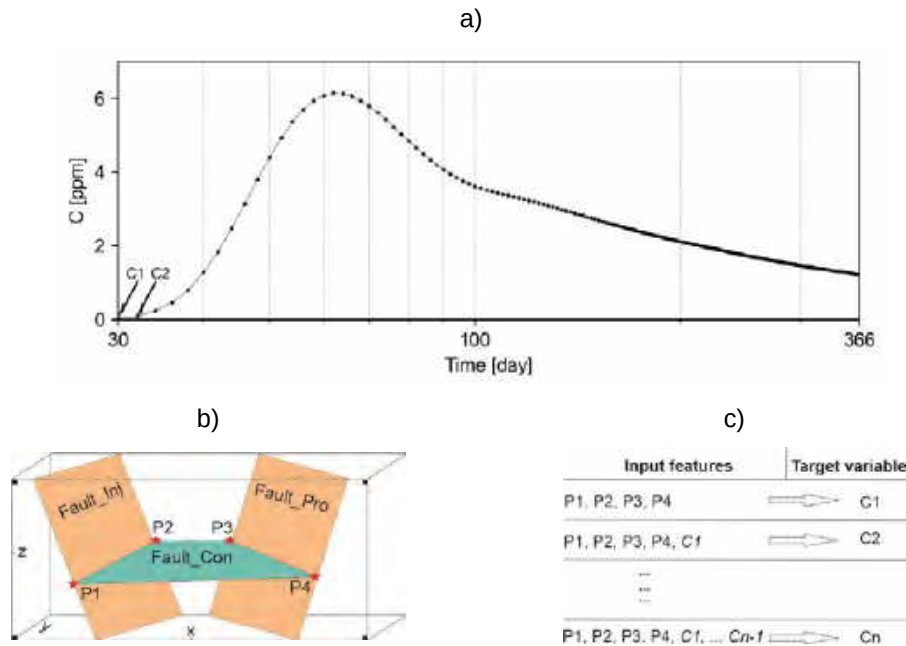
- Strategy 1: Two ML models, DTR and RFR, are trained to independently predict the tracer concentration values. Both models predict the entire time steps of the BTC, using the input features. In this study DTR and RFR correlate structural information of the geological model with the tracer concentration. While in DTR a single tree is trained to capture the relation between the input features and target variable, RFR cultivates several trees in parallel (bagging). DTR is simple to implement and interpret, but it can be prone to overfitting. Therefore, the more complex RFR is also included in this study. The mathematical foundations of DTR and RFR are well-documented in the literature e.g. Kotsiantis (2013), Liu et al. (2012) and XU et al. (2005).
- Strategy 2: A GBR model is used to predict the concentration value at each time step by correlating it with the previous prediction. The GBR is an ensemble method that combines multiple simple and weak learners sequentially (bagging) to improve the overall performance of the model. This approach, denoted as the chain model, requires GBR to be executed for each time step of the BTC. Details of this approach are elaborated in the following.

127 2.2.1 Chain GBR model

128 Figure 3 provides an overview of the chain regression model for the simple case. A BTC,
129 serving as the target variable, is presented in Figure 3-a. The input features are composed of
130 the structural geometric information from the reservoir model with the coordinates of four
131 corners of the uncertain sub-horizontal fracture (P1, P2, P3, and P4 in Figure 3-b). The model
132 correlates the x/z coordinates with the BTC concentration values, i.e. the y-coordinate data
133 remain fixed across all scenarios for the sake of simplicity. All the governing equations and
134 modelling assumptions behind the calculation of the BTCs are fully addressed in Dashti et al.
135 (2023). For the complex case, coordinates of the two uncertain fracture surfaces are used as
136 the input feature while the BTC data are target variables.

137 The chain model predicts the BTC concentration values in a sequential manner. It starts by
138 predicting the concentration for the first time step (C1) based on the input features (Figure 3-b
139 and c). For the second and following time steps (C2), the model uses the previous values, i.e.
140 C1, along with the input features. Some errors can exist in the predicted C1 by GBR. However,
141 to predict C2, the input feature list still contains 8 coordinate values than have a higher impact
142 compared to the recently predicted C1. This gradual addition of the predicted values can help
143 the chain model to adjust the weight of added features, i.e. previously predicted concentrations.
144 Figure 3-c illustrates how concentration values from previous steps concatenate in the input
145 features' list. To predict the first concentration value (C1) in the GBR chain model, the input
146 feature list initially contains eight values. To predict the concentration for the last time step of
147 the simple case (C169), the input feature list contains eight coordinates and 168 previously
148 predicted concentration values. In the complex case, the BTC includes 140 concentration
149 values. The input feature list of the DTR and RFR models remains fixed, because these two
150 methods predict all the time steps of the BTC merely based on the coordinates of the fractures.

151



152

153 Figure 3. Workflow developed for chain GBR model a) A BTC representing concentration values, C, versus
 154 logarithmic time scale. b) Four corners of the sub-horizontal uncertain fault, P1, P2, P3, and P4, are used in the
 155 ML model to predict the first concentration value (C1) for the simple case. c) To predict the second concentration
 156 value (C2), the first predicted value (C1) is also included besides the coordinates of four corners. In each time
 157 step, the previous values are added up to the list of input features.

158 The GBR algorithm (Friedman, 2002) is selected due to its simplicity, bagging nature, and
 159 efficiency as a predictor for the chain model. Like other supervised ML algorithms (Gupta,
 160 2022), GBR learns a function that maps the input feature/s (x) to target variable/s $f(x)$ with
 161 the minimum loss:

$$162 \quad F(x) = \operatorname{argmin}_L \left(f(x), \hat{f}(x) \right) \text{ Eq. 1}$$

163 where L is the loss function and $\hat{f}(x)$ represents the prediction. The loss function is chosen
 164 based on the type of learning (e.g., regression, classification) and the type of the target variable
 165 (e.g., discrete, continuous). Squared error (L_2) loss (Bühlmann and Yu, 2003) is a simple and
 166 efficient loss function when outliers are not expected and is hence chosen here:

$$167 \quad L = \sum_{i=1}^n \frac{1}{2} \left(f(x) - \hat{f}(x) \right)^2 \quad \text{Eq. 2}$$

168 ML methods generating an ensemble of predicting models in parallel (bagging methods like
 169 RFR) or sequential (boosting methods like GBR) are more reliable than models consisting of
 170 a single strong predictive model (like DTR) (Fanelli et al., 2013; Shu and Burn, 2004). Boosting
 171 methods like GBR can have a better performance for working on small data sets compared to
 172 bagging methods that distribute the data set between different predictors. GBR starts with a
 173 very simple model ($F_0(x)$), trying to fit a straight horizontal line (average of target variable). In
 174 fact, the derivation of the loss function with respect to the predictions establishes the average

value as the best guess for the first tree. In the next round, the GBR algorithm maps the input features to the residuals (remaining errors) of the previous tree, a process that can be interpreted as performing gradient descent on the negative derivative of the difference between prediction and target variable w.r.t. the prediction (Breiman, 1998). The use of residuals rather than absolute values is another reason for choosing GBR. This allows for the inclusion of residuals contributed by recently error-prone predicted concentration values into the model. In subsequent rounds, new decision trees are trained based on the accumulated residuals of the whole ensemble (Schapire, 2003):

$$\hat{F}_m(x) = F_{m-1}(x) + \alpha_m \hat{f}_m(x) \quad \text{Eq. 3}$$

where $\hat{F}_m(x)$ represents the final general function that connects input features to the target variable, $F_{m-1}(x)$ contains the information from all previous trees, α is the learning rate that avoids overfitting and $\hat{f}_m(x)$ represents the last tree that is correlating remaining residuals and the input features. Low learning rates decrease the impact of each tree, i.e., more trees will be needed but the model also will be more generalized. GBR minimizes the error of each tree and uses the remaining errors as the target variable of the next tree. In this way, the model is trained based on its minimized errors and aggregates several trees with decreasing errors. He et al. (2022) delved into the details of the GBR.

2.2.2 ML model optimization and quality control

Each ML model has two types of arguments: 1) inputs that include hyperparameters (parameters related to the model's architecture) and features selected by the user for predicting the target variable/s, and 2) output arguments that consist of internal weights and the target variable/s. The ML model is trained to minimize the error by tuning its input arguments, allowing the learning algorithm to optimize the output arguments and achieve better scores on the withheld test set (Alpaydin, 2020; Hutter et al., 2019). This iterative process, known as hyperparameter tuning (Raschka and Mirjalili, 2019) involves optimization of parameters such as the learning rate, number of trees, maximum depth of trees, etc. to decrease the error. Determining the optimal number of trees poses a challenge due to the bias-variance trade-off (Oshiro et al., 2012; Probst et al., 2019). Another hyperparameter, the maximum depth of a tree, is defined as the longest path between the root node (first node) and the leaf node (last node).

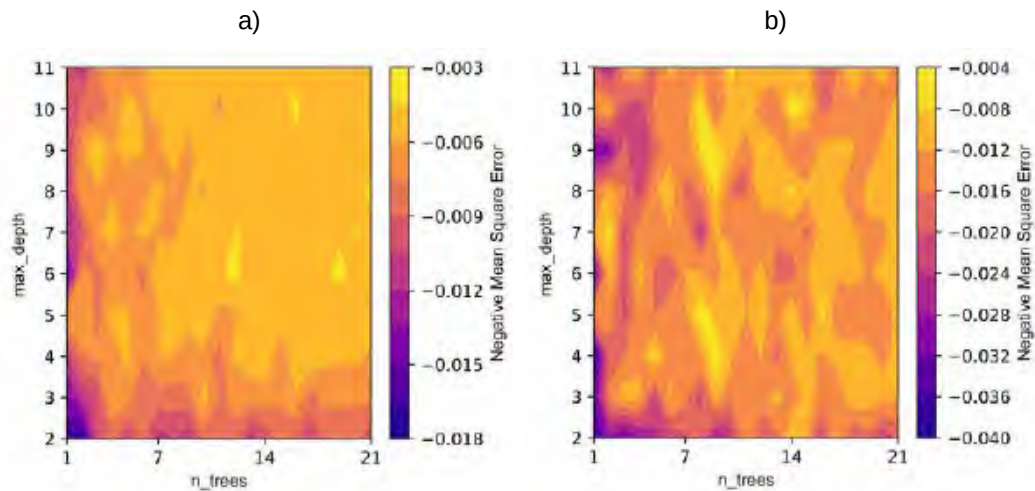
Grid search is a hyperparameter tuning method that allows input arguments to be defined as a range rather than a single value. It performs an exhaustive search over all possible combinations of values to identify the model with the lowest error i.e. highest score. For the RFR model, the number of trees and maximum depth is considered as arrays with 20 and 10 elements, respectively that result in 200 combinations. For the DTR model also maximum depth of each tree, the minimum number of samples in a leaf node and the minimum number of samples for splitting an internal node are tuned. In total, an ensemble of 540 models has

212 been calculated using hyperparameter tuning for the DTR method. In the GBR algorithm of the
213 chain model default values are used. Conventionally, higher score values are preferred over
214 lower ones, and therefore we also tried to find out the combination with the maximum negated
215 mean squared error (MSE) using the grid search.

216 To evaluate the model's performance, k-fold cross-validation (Zhang et al., 1999) has been
217 employed. Rather than splitting the input data into train and test, it randomly splits them
218 arbitrarily into k number of "splits". Then, the ML model will keep one split as the test and all
219 others as the train sets. In the case of splitting data into five splits, the same number of models
220 will be run and in each run, splits will be shuffled. This five-run procedure will be performed for
221 all the assumed 200 combinations of hyperparameters in the grid search for the RFR method.
222 Therefore, it finally creates 1'000 ML models – each of them being an ensemble of individual
223 trees – and the ensemble with the highest score will be used for the final prediction. In this
224 study, we follow the recommendations in the literature (An et al., 2007; Erdogan Erten et al.,
225 2022) and use five splits for cross-validation for all three methods. Training (online) time for
226 the 1,000 ML models of the RFR model on a Core i7 laptop is approximately 10 seconds. For
227 DTR, with an ensemble of 2,700 ML models, the online time remains to be around 10 seconds.
228 The simplicity of DTR compared to RFR results in faster computation. The chain model proves
229 to be the most time-consuming approach, taking around 70 seconds for training without any
230 hyperparameter optimization. Several hyperparameters were tested for the chain model, but
231 the online time only increased without improving the model's accuracy. Therefore, default
232 values were chosen for the chain model. For both the simple and complex cases several values
233 have been tried for the learning rate in hyperparameter tuning but in the end, the default one
234 (0.1) has been used. The required time for predicting a new solution with the trained models
235 (offline time) remains in the range of milliseconds. To access the input data and trained ML
236 models of two cases, please refer to the code and data availability section.

237 Figure 4-a and b show the distribution of the negative MSE scoring metric in train and test
238 splits, focusing on two tuned hyperparameters of the RFR model in the simple case. The
239 average of the MSE in the four train splits is presented in Figure 4-a. The distribution of the
240 average scoring metric in the train splits is influenced by both the number and maximum depth
241 of trees. Based on Figure 4-a, the accuracy of the model increases as both the maximum depth
242 of trees and the number of trees increase. However, the score distribution in the test split
243 (Figure 4-b) is more complicated. The scores in the test split are generally lower than those in
244 the train splits (-0.04 to -0.004 versus -0.018 to -0.003). While the score distribution for the
245 train split promises high accuracy of the model by increasing the two hyperparameters, the
246 heterogeneous distribution in Figure 4-b raises doubts on this conclusion. The presented
247 example in Figure 4 concludes that determining the optimal combination even for only two
248 hyperparameters is not a straightforward task. Going to higher dimensions can make the

249 situation more complicated and unsolvable. Therefore, methods like grid search identify the
 250 best combination of tuned hyperparameters.



251 Figure 4. Change in the accuracy of the ML model with respect to different combinations of two hyperparameters
 252 of the RFR model on the train (a) and test (b) splits. The accuracy distribution in the train split (a) is smooth and
 253 higher accuracies can be achieved by increasing the number of trees and maximum depth of each tree. Subplot
 254 b depicts the more patchy and anisotropic behavior of the accuracy with respect to the hyperparameters.

255 3 Results

256 3.1 Simple case

257 Dashti et al. (2023) employed numerical simulations to assess the effects of uncertainty in
 258 structural models using 50 different structural scenarios in a simplified EGS setting. In these
 259 synthetic models, a 24-hour tracer injection on day eight of the simulation was assumed and
 260 monitored along a one-year time span in the production well (see Figure 5-a with e.g. peak
 261 concentration time varying between days 54 and 68). To better present the variations, a box
 262 plot (Figure 5-b) is generated by extracting the highest concentration value from each BTC and
 263 normalizing them based on their median. The variation of the tracer peak concentration time,
 264 as well as a 25% fluctuation in peak magnitude, emphasize the significance of structural
 265 uncertainty, which can introduce unexpected deviations in the results of important field tests.
 266 The appearance of a second peak between days 100 and 150 in Figure 5-a is due to the
 267 reinjection of the tracer, not multiple flow paths or stagnation zones. The first 30 days of the
 268 simulation are disregarded due to negligible concentration (almost zero) of the tracer in the
 269 production well during that period.

270

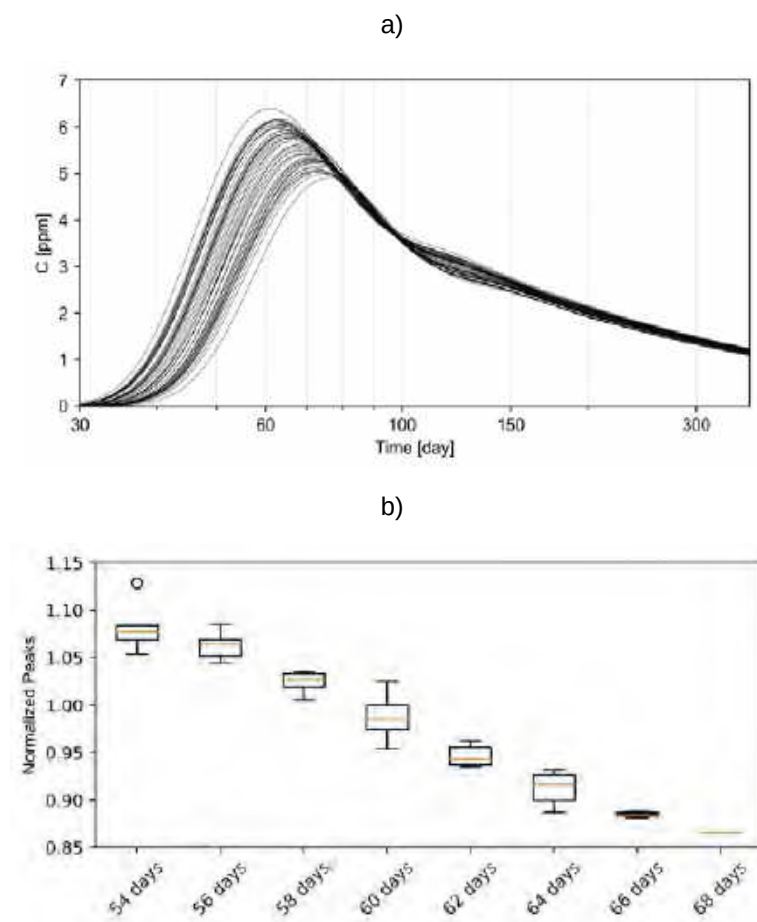
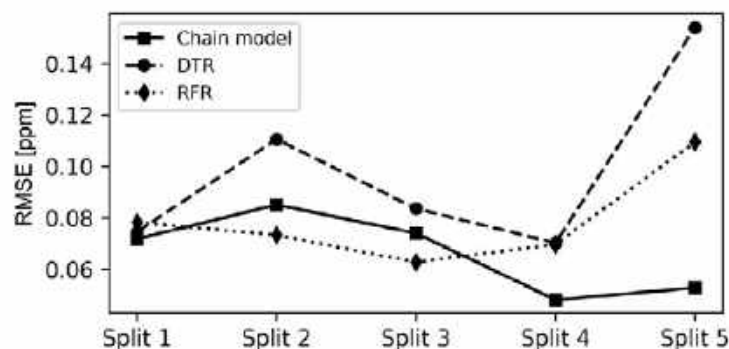


Figure 5. a) Unique BTCs simulated using the finite element solver and used as target variables for the ML models. BTCs are different from each other due to changing structural models. b) A box plot visualizing the normalized peak concentration values versus the time of the calculated peak (analysis based on Dashti et al. (2023)).

Results of the k-fold cross-validation in Figure 6 show how RMSE varies in five splits of the three ML methods. The average RMSE of the chain model is lower than the DTR and RFR. Apart from the higher absolute accuracy, the homogeneity of the model's performance is another important factor to consider. Based on Figure 6, RMSE values in the DTR model show higher standard deviations. The higher standard deviation of RMSE for the DTR model suggests that it is overfitting the training data. Overfitting occurs when a model learns the training data too well and is unable to generalize to new data. In the case of the DTR model, this may be due to the fact that it is a single-tree model. Hence it is more likely to memorize the training data than an ensemble model like the RFR or chain model. In this study, the simplicity of the DTR model is the main factor leading to overfitting issues. The RFR model mitigates overfitting by initiating multiple parallel trees that distribute the input data. The chain model also incorporates several sequential models that consistently outperform a single model. Overall, the chain model is the most accurate and robust ML model for predicting BTCs

in cases without any historical data. It has a lower average RMSE and a lower standard deviation of RMSE than the RFR and DTR models.



290

291 Figure 6. Accuracy distribution of the three designed ML models within their splits. RMSE values are represented
292 as accuracy parameters.

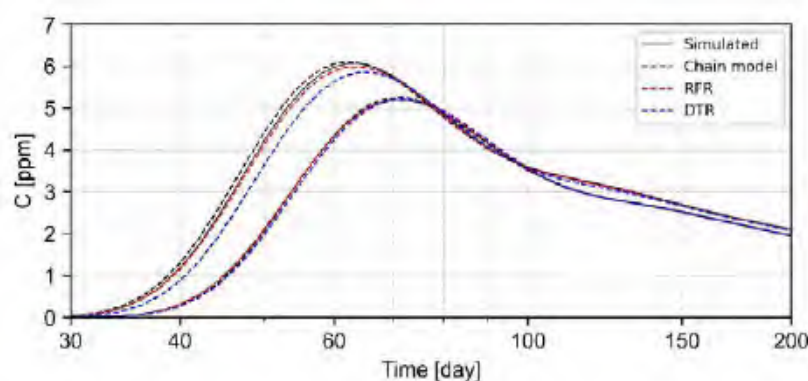
293 To better assess the trained models and prevent information leakage, two additional scenarios
294 are imported into the three ML models. The trained ML models are then utilized to predict the
295 BTCs of these two new test scenarios. Table 1 presents the accumulated RMSEs of these
296 two test scenarios (test set) and models' input data (train set). The ML models exhibit an
297 increase in error when transitioning from train to test scenarios. However, even for the two new
298 test scenarios, the RMSE remains at an acceptable level. The DTR model had the largest
299 difference in RMSE between the train and test sets, which clearly indicates overfitting. The
300 RFR and the chain models yield a better balance in terms of RMSE between the train and test
301 data, suggesting their improved performance and ability to generalize.

302 Table 1. RMSE values of the three designed ML models within the train and test sets.

	DTR	RFR	chain model
train set	1.1×10^{-4}	1.0×10^{-4}	1.2×10^{-4}
test set	1.5×10^{-1}	4.0×10^{-2}	5.2×10^{-2}

303

304 Figure 7 shows the numerically simulated BTCs of two test scenarios and the outputs of three
305 ML methods. For one of the test scenarios, all three ML methods achieved similar and reliable
306 results compared to the simulation results. For the other test scenario, the DTR method was
307 less accurate than the other methods, likely due to overfitting. The RFR and chain models had
308 similar levels of accuracy.



309

310 Figure 7. Two different test cases were investigated to understand the accuracy of ML models. The chain model
 311 and RFR have a high accuracy in both cases.

312 To further evaluate the trained models, an additional set of 2'000 different structural scenarios
 313 is generated and imported into ML models. In this step, only the connecting fault is perturbed,
 314 and the coordinates of its four corners are inputted into the three ML models. Figure 8 provides
 315 a visualization of the BTCs generated by the three ML models. These 6'000 BTCs presented
 316 in Figure 8 are calculated in the scale of milliseconds using DTR, RFR, and chain models. Two
 317 extreme cases from the training data are highlighted with blue color and dots to illustrate the
 318 boundaries of expectations. The RFR method perfectly follows the trend, generating 2'000
 319 almost unique and parallel BTCs (Figure 8-a), which suggests that it may be underfitting the
 320 training data. The underfit models have a high bias due to oversimplifications and ignoring the
 321 underlying patterns in the train data. This problem can directly originate from the insufficient
 322 input data used to train the RFR model. The bagging procedure of RFR splits 50 input data
 323 sets into parallel bags making it difficult for each tree to be a balanced predictor. On the other
 324 hand, DTR has generated far fewer unique BTCs as shown in Figure 8-b. The covered area
 325 with BTC curves in Figure 8-a and b differs dramatically. DTR mainly repeats what it has
 326 observed in the training step. As Figure 9 shows, only a few new BTCs are generated and the
 327 majority of 2'000 BTCs overlap the 50 BTCs used in the training step.

328 The chain model consistently generated more reliable BTCs compared to RFR and DTR
 329 (Figure 8-c). However, in some cases, the chain model generated BTCs with irregular patterns,
 330 such as concentration values fluctuating around the peak. Despite these local discrepancies,
 331 the chain model is still the most reliable ML model for predicting BTCs.

332 Another notable point is that all the three data-driven ML methods are unable to be used for
 333 extrapolation. Even the frequency of generated BTCs decreases close to the extreme point for
 334 three subplots shown in Figure 8. This issue is the worst with the DTR method while the chain
 335 model has generated more BTCs in the adjacency of the extreme cases.

336

337

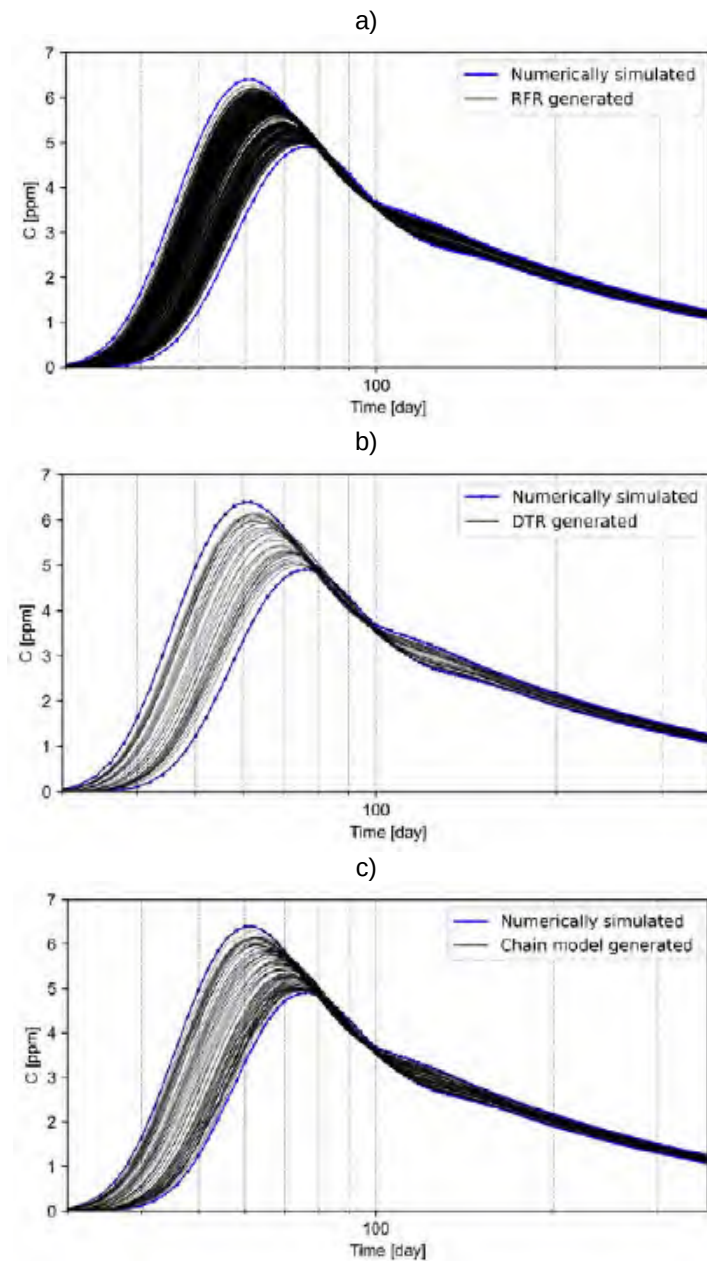
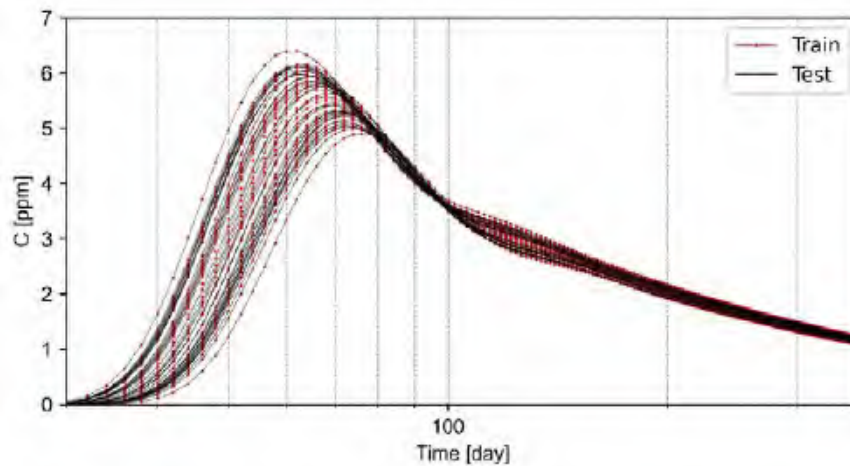


Figure 8. Two thousand generated BTCs using RFR (a), DTR (b), and chain model (c). Two extreme cases coming from the simulation are highlighted as blue curves with dots.



340

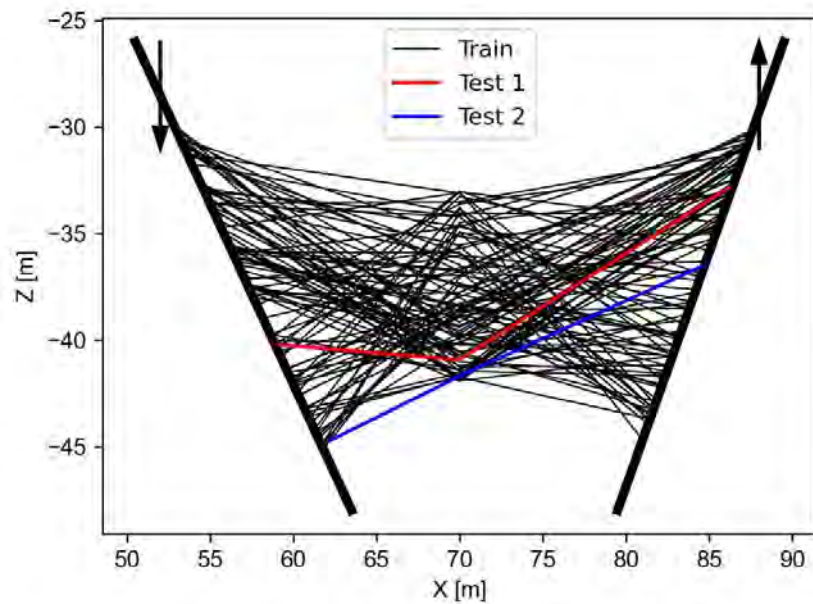
341 Figure 9. Most of the 2'000 BTCs generated by DTR (named Test and shown as solid black line) exactly match
 342 the input data used for training the model.

343 3.2 Complex case

344 For the complex case, 100 BTCs are simulated in the numerical solver and used to train and
 345 test the three ML models. The number of scenarios has increased compared to the simple
 346 case (with 50 simulations) due to the complexity of the model. In the complex case, a 24-hour
 347 tracer injection on day five of the simulation is assumed and monitored for two months in the
 348 production well. The simulation time is decreased due to the shorter/faster connection between
 349 the injection and production wells. Figure 10 shows a 2D section of the 100 unique pathways
 350 that connect injection and production wells. Two pathways are plotted with red and blue colors
 351 and are used to test the validity of the ML methods because they have not been used in the
 352 training process. Test 1 scenario visually demonstrates how the two connecting fractures can
 353 have different depths and dipping angles.

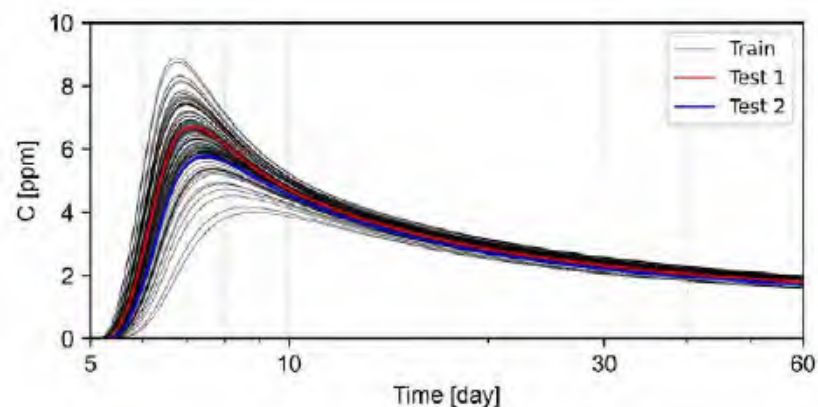
354 Figure 11 shows the numerically simulated BTCs for 100 scenarios of the complex case.
 355 Similar to the simple case, the peak concentration time and magnitude of the BTCs vary due
 356 to the change in the geometrical properties of the fracture network. The color coding of the
 357 train and test scenarios (1 and 2) remains consistent with Figure 10.

358



359

360 Figure 10. A 2D cross-section from the middle of the complex model. Thin black lines represent the trace of the
 361 two uncertain fractures that connect certain fractures shown via two thick black lines. The red and blue traces
 362 represent the geometry of the uncertain fractures in two tests. Arrows show the location of the injection and
 363 production wells.

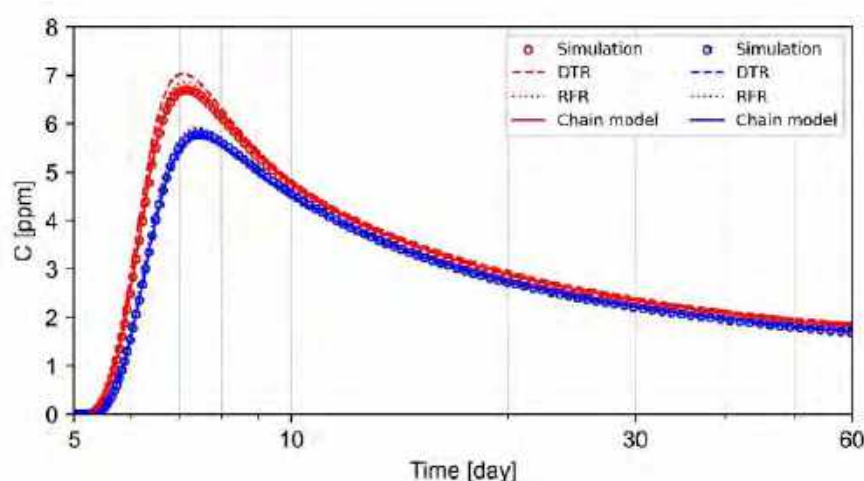


364

365 Figure 11. Thin black curves represent 98 BTCs simulated using the finite element solver. Two test scenarios are
 366 also named as Test 1 and Test 2. To see the geological model of the test cases refer to Figure 10.

367 Two test scenarios of the ML methods are shown in Figure 12. The RMSE values confirm the
 368 higher accuracy of the chain method. The cumulative RMSE for both scenarios is 0.05 ppm
 369 for the chain model, 0.18 ppm for DTR, and 0.16 ppm for RFR. Notably, all three machine
 370 learning models were employed with the same hyperparameters for both the simple and
 371 complex cases.

372



373

374 Figure 12. Simulation and ML-generated results for Test 1 are plotted as red circles and lines. Results related to
 375 Test 2 are plotted as blue circles and lines.

376 4 Discussion

377 Detailing the observed errors is crucial for future work aimed at improving the interpretability
 378 of the ML methods' performance. The (negligible) discrepancy likely stems from the distribution
 379 of test scenarios and size of input data (50 and 100 scenarios). This finding underscores the
 380 sensitivity of data-driven models to input data distribution. As extrapolation is a known
 381 challenge for such models, selecting a test sample near the boundary in this study exemplifies
 382 this limitation. A uniform high-density sampling strategy may prove more effective than the
 383 Gaussian distribution.

384 Even with large datasets, data-driven ML methods can still deviate from the underlying physics.
 385 Degen et al. (2023) proposed promising physics-based ML methods using order reduction
 386 techniques e.g., non-intrusive reduced basis, to build the solution based on basis functions
 387 that preserve the structure of the physics. In this study, we employed a sequence of
 388 concentration values as the target variable, allowing the ML models to learn the temporal
 389 relationships. Three tested ML methods have been able to capture the trend for two different
 390 cases. The current limitation is that the concentration prediction is restricted to a single point
 391 within the model. However, our strategy can be extended to develop ML models that predict
 392 the target variable at various points over time.

393 Meanwhile, the ML methods were significantly faster than the numerical solver, with up to six
 394 orders of magnitude reduction in computational time. To numerically solve the problem of the
 395 simple case, 12 cores on a high-performance computing cluster should run for 4 hours. The
 396 whole time for constructing (offline) and applying (online) the ML models remains in the scale
 397 of seconds. This substantial reduction makes uncertainty analysis feasible using fast and
 398 reliable ML models, without relying on time-consuming simulations that typically span multiple
 399 days. This concept can also be suited for including structural uncertainties in more complicated

400 EGS settings with several intersecting fractures.

401 **5 Conclusion and outlook**

402 This study presents a novel approach for using ML methods that enables quantifying the
403 impact of structural uncertainty on BTCs in two EGS reservoirs. The approach was the first
404 test to expand the range of structural reservoir models using ML techniques, based on an
405 original set of a limited number of the numerical scenarios. This meets the specific requirement
406 of uncertainty quantification, which is to provide a broad range of scenarios.

407 Different ML approaches are trained using the available numerical simulations to predict the
408 BTCs based on the geometries of the perturbed elements. One ML approach used DTR and
409 RFR algorithms to predict the entire BTC at once. Another ML approach employed a chain of
410 GBR models to predict each time step of the BTCs while considering the correlation between
411 consecutive time steps. The DTR model suffered from overfitting, while the RFR and chain
412 models were more reliable, achieving an acceptable accuracy with a balanced accumulated
413 RMSE in train and test scenarios. In the simple case, the RMSE for the DTR model jumped
414 from 0.00011 to 0.15 between train and test scenarios, while for the RFR and chain models, it
415 reached from 0.0001 to 0.04 and from 0.00012 to 0.052, respectively.

416 The trained ML models are further applied to generate BTCs for 2'000 unique structural
417 scenarios in the model with a simple geometry. The chain model was more accurate than the
418 RFR and DTR models. The RFR method produced 2'000 BTCs that closely follow the trend
419 observed in the training set indicating the underfitting issue, whereas DTR can only replicate
420 the BTCs from the training set. The chain model captures both the general trend and small-
421 scale patterns of the data. However, the accuracy and reliability in all three methods decreases
422 for test cases that are close to the boundaries of the input test data. A uniform sampling for
423 selecting the input data can help the ML methods to have a wide and homogeneous distribution
424 in the test data.

425 The presented approach can be adopted for a broader number of forward calculation schemes.
426 This opens up new possibilities for more complex fractured rock settings. Rather than
427 coordinates of one/two fractures, a more complex structural network from a real-world EGS
428 case can be used as the input features for the ML methods.

429 While only structural models were varied herein to assess their impact on the BTCs, future
430 applications could encompass modifications to specific petrophysical properties of the
431 reservoir, further expanding the possibilities of stochasticity. Conversely, integrating more data
432 into the model, such as BTC's or hydraulic testing data obtained from specific EGS well
433 configurations (e.g. Schill et al., 2017), can reduce structural uncertainties. This allows for the
434 rapid elimination of non-viable models using ML-driven routines.

435 Harnessing the computational efficiency of ML, this innovative approach can be transformed
436 into a surrogate model, effectively representing the core of an inverse, backward calculation

scheme for parameter identification. This transformation has the potential to replace conventional analytical solutions, which are currently the primary method for estimating parameters from tracer campaigns. The ML-based surrogate model offers several advantages, including significantly faster calculation speeds and the ability to capture the non-uniqueness inherent in mathematical solutions. In this framework, BTC data serve as the primary input, while the parameters of the complex EGS reservoir represent the target variables.

Competing interests

The authors declare that they have no conflict of interest.

Acknowledgements

Ali Dashti is receiving the financial support from The German Academic Exchange Service (Deutscher Akademischer Austauschdienst: DAAD) to do his PhD in Germany as the Research Grants-Doctoral programmes in Germany 2019/20. This organization is appreciated for giving the opportunity to researchers. The study is also part of the Helmholtz portfolio project Geoenergy. The support from the program “Renewable Energies”, under the topic “Geothermal Energy Systems”, is gratefully acknowledged. Two anonymous reviewers are appreciated for their insightful comments.

Code and data availability

Required data and developed ML methods for the simple and complex cases are documented and available in a Zenodo repository (<https://zenodo.org/records/10810243>).

6 References

- Abbaszadeh Shahri, A., Shan, C., Larsson, S., 2022. A Novel Approach to Uncertainty Quantification in Groundwater Table Modeling by Automated Predictive Deep Learning. *Nat Resour Res* 31 (3), 1351–1373.
- Alakeely, A., Horne, R.N., 2020. Simulating the Behavior of Reservoirs with Convolutional and Recurrent Neural Networks. *SPE Reservoir Evaluation & Engineering* 23 (03), 992–1005.
- Alpaydin, E., 2020. Introduction to machine learning, Fourth edition ed. Adaptive Computation and Machine Learning. The MIT Press, Cambridge, London, XXIV, 682, [2] strony.
- An, S., Liu, W., Venkatesh, S., 2007. Fast cross-validation algorithms for least squares support vector machine and kernel ridge regression. *Pattern Recognition* 40 (8), 2154–2162.
- Bauer, P., Thorpe, A., Brunet, G., 2015. The quiet revolution of numerical weather prediction. *Nature* 525 (7567), 47–55.
- Benoit, A., Lefèvre, L., Orgerie, A.-C., Raïs, I., 2018. Reducing the energy consumption of large-scale computing systems through combined shutdown policies with multiple constraints. *The International Journal of High Performance Computing Applications* 32 (1), 176–188.
- Bond, C.E., 2015. Uncertainty in structural interpretation: Lessons to be learnt. *Journal of Structural Geology* 74, 185–200.
- Borgonovo, E., Plischke, E., 2016. Sensitivity analysis: A review of recent advances. *European Journal of Operational Research* 248 (3), 869–887.
- Breiman, L., 1998. Arcing classifier (with discussion and a rejoinder by the author). *Ann. Statist.* 26 (3).

- 477 Brunton, S.L., Kutz, J.N., 2022. Data-driven science and engineering: Machine learning, dynamical
478 systems, and control. Cambridge University Press, Cambridge, United Kingdom, New York, NY,
479 pages cm.
- 480 Bühlmann, P., Yu, B., 2003. Boosting With the L2 Loss. *Journal of the American Statistical Association*
481 98 (462), 324–339.
- 482 Cao, V., Schaffer, M., Taherdangkoo, R., Licha, T., 2020. Solute Reactive Tracers for Hydrogeological
483 Applications: A Short Review and Future Prospects. *Water* 12 (3), 653.
- 484 Carleo, G., Cirac, I., Cranmer, K., Daudet, L., Schuld, M., Tishby, N., Vogt-Maranto, L., Zdeborová, L.,
485 2019. Machine learning and the physical sciences. *Rev. Mod. Phys.* 91 (4).
- 486 Dashti, A., Gholami Korzani, M., Geuzaine, C., Egert, R., Kohl, T., 2023. Impact of structural
487 uncertainty on tracer test design in faulted geothermal reservoirs. *Geothermics* 107, 102607.
- 488 Degen, D., Caviedes Voullième, D., Buitier, S., Hendriks Franssen, H.-J., Vereecken, H., González-
489 Nicolás, A., Wellmann, F., 2023. Perspectives of Physics-Based Machine Learning for Geoscientific
490 Applications Governed by Partial Differential Equations.
- 491 Doost, S.N., Ghista, D., Su, B., Zhong, L., Morsi, Y.S., 2016. Heart blood flow simulation: a perspective
492 review. *Biomedical engineering online* 15 (1), 101.
- 493 Egert, R., Korzani, M.G., Held, S., Kohl, T., 2020. Implications on large-scale flow of the fractured EGS
494 reservoir Soultz inferred from hydraulic data and tracer experiments. *Geothermics* 84, 101749.
- 495 Erdogan Erten, G., Yavuz, M., Deutsch, C.V., 2022. Combination of Machine Learning and Kriging for
496 Spatial Estimation of Geological Attributes. *Nat Resour Res* 31 (1), 191–213.
- 497 Fanelli, G., Dantone, M., Gall, J., Fossati, A., van Gool, L., 2013. Random Forests for Real Time 3D Face
498 Analysis. *Int J Comput Vis* 101 (3), 437–458.
- 499 Friedman, J.H., 2002. Stochastic gradient boosting. *Computational Statistics & Data Analysis* 38 (4),
500 367–378.
- 501 Gudmundsdottir, H., Horne, R., 2020. Prediction Modeling for Geothermal Reservoirs Using Deep
502 Learning. *PROCEEDINGS, 45th Workshop on Geothermal Reservoir Engineering*.
- 503 Gupta, M., 2022. A Comparative Study on Supervised Machine Learning Algorithm. *IJRASET* 10 (1),
504 1023–1028.
- 505 He, J., Li, K., Wang, X., Gao, N., Mao, X., Jia, L., 2022. A Machine Learning Methodology for Predicting
506 Geothermal Heat Flow in the Bohai Bay Basin, China. *Nat Resour Res* 31 (1), 237–260.
- 507 Hutter, F., Kotthoff, L., Vanschoren, J., 2019. *Automated Machine Learning*. Springer International
508 Publishing, Cham.
- 509 Karniadakis, G.E., Kevrekidis, I.G., Lu, L., Perdikaris, P., Wang, S., Yang, L., 2021. Physics-informed
510 machine learning. *Nat Rev Phys* 3 (6), 422–440.
- 511 Kharazmi, E., Zhang, Z., Karniadakis, G.E., 2019. Variational Physics-Informed Neural Networks For
512 Solving Partial Differential Equations.
- 513 Knapp, E., Battaglia, M., Stadelmann, T., Jenatsch, S., Ruhstaller, B., 2021. XGBoost Trained on
514 Synthetic Data to Extract Material Parameters of Organic Semiconductors, in: *2021 8th Swiss*
515 *Conference on Data Science (SDS)*. 2021 8th Swiss Conference on Data Science (SDS), Lucerne,
516 Switzerland. 6/9/2021 - 6/9/2021. IEEE, pp. 46–51.
- 517 Kohl, T., Mégel, T., 2007. Predictive modeling of reservoir response to hydraulic stimulations at the
518 European EGS site Soultz-sous-Forêts. *International Journal of Rock Mechanics and Mining*
519 *Sciences* 44 (8), 1118–1131.
- 520 Kotsiantis, S.B., 2013. Decision trees: a recent overview. *Artif Intell Rev* 39 (4), 261–283.
- 521 Liu, Y., Wang, Y., Zhang, J., 2012. New Machine Learning Algorithm: Random Forest, in: Hutchison, D.,
522 Kanade, T., Kittler, J., Kleinberg, J.M., Mattern, F., Mitchell, J.C., Naor, M., Nierstrasz, O., Pandu
523 Rangan, C., Steffen, B., Sudan, M., Terzopoulos, D., Tygar, D., Vardi, M.Y., Weikum, G., Liu, B., Ma,

- 524 M., Chang, J. (Eds.), Information Computing and Applications, vol. 7473. Lecture Notes in
525 Computer Science. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 246–252.
- 526 Meakin, P., Tartakovsky, A.M., 2009. Modeling and simulation of pore-scale multiphase fluid flow and
527 reactive transport in fractured and porous media. *Rev. Geophys.* 47 (3).
- 528 Okoroafor, E.R., Smith, C.M., Ochie, K.I., Nwosu, C.J., Gudmundsdottir, H., Aljubran, M., 2022.
529 Machine learning in subsurface geothermal energy: Two decades in review. *Geothermics* 102,
530 102401.
- 531 Oshiro, T.M., Perez, P.S., Baranauskas, J.A., 2012. How Many Trees in a Random Forest?,
532 in: Hutchison, D., Kanade, T., Kittler, J., Kleinberg, J.M., Mattern, F., Mitchell, J.C., Naor, M.,
533 Nierstrasz, O., Pandu Rangan, C., Steffen, B., Sudan, M., Terzopoulos, D., Tygar, D., Vardi, M.Y.,
534 Weikum, G., Perner, P. (Eds.), Machine Learning and Data Mining in Pattern Recognition, vol.
535 7376. Lecture Notes in Computer Science. Springer Berlin Heidelberg, Berlin, Heidelberg, pp.
536 154–168.
- 537 Probst, P., Wright, M.N., Boulesteix, A.-L., 2019. Hyperparameters and tuning strategies for random
538 forest. *WIREs Data Mining Knowl Discov* 9 (3).
- 539 Raissi, M., Perdikaris, P., Karniadakis, G.E., 2019. Physics-informed neural networks: A deep learning
540 framework for solving forward and inverse problems involving nonlinear partial differential
541 equations. *Journal of Computational Physics* 378, 686–707.
- 542 Raschka, S., Mirjalili, V., 2019. Python machine learning: Machine learning and deep learning with
543 python, scikit-learn, and tensorflow 2, Third edition ed. Packt Publishing, Limited, Birmingham,
544 741 pp.
- 545 Schapire, R.E., 2003. The Boosting Approach to Machine Learning: An Overview, in: Bickel, P., Diggle,
546 P., Fienberg, S., Krickeberg, K., Olkin, I., Wermuth, N., Zeger, S., Denison, D.D., Hansen, M.H.,
547 Holmes, C.C., Mallick, B., Yu, B. (Eds.), Nonlinear Estimation and Classification, vol. 171. Lecture
548 Notes in Statistics. Springer New York, New York, NY, pp. 149–171.
- 549 Schill, E., Genter, A., Cuenot, N., Kohl, T., 2017. Hydraulic performance history at the Soultz EGS
550 reservoirs from stimulation and long-term circulation tests. *Geothermics* 70, 110–124.
- 551 Shu, C., Burn, D.H., 2004. Artificial neural network ensembles and their application in pooled flood
552 frequency analysis. *Water Resour. Res.* 40 (9).
- 553 Soize, C., 2017. Uncertainty Quantification 47. Springer International Publishing, Cham.
- 554 Stadelmann, T., Tolkachev, V., Sick, B., Stampfli, J., Dürr, O., 2019. Beyond ImageNet: Deep Learning
555 in Industrial Practice, in: Braschler, M., Stadelmann, T., Stockinger, K. (Eds.), Applied Data Science.
556 Springer International Publishing, Cham, pp. 205–232.
- 557 Weigend, A.S., Gershenfeld, N.A. (Eds.), 1994. Time series prediction: Forecasting the future and
558 understanding the past : proceedings of the NATO Advanced Research Workshop on Comparative
559 Time Series Analysis, held in Santa Fe, New Mexico, May 14-17, 1992. A Proceedings volume,
560 Santa Fe Institute studies in the sciences of complexity XV. Routledge, Taylor & Francis Group,
561 New York, NY, 643 pp.
- 562 Wellmann, J.F., Horowitz, F.G., Schill, E., Regenauer-Lieb, K., 2010. Towards incorporating uncertainty
563 of structural data in 3D geological inversion. *Tectonophysics* 490 (3-4), 141–151.
- 564 XU, M., WATANACHATURAPORN, P., VARSHNEY, P., ARORA, M., 2005. Decision tree regression for
565 soft classification of remote sensing data. *Remote Sensing of Environment* 97 (3), 322–336.
- 566 Yu, J., Lu, L., Meng, X., Karniadakis, G.E., 2022. Gradient-enhanced physics-informed neural networks
567 for forward and inverse PDE problems. *Computer Methods in Applied Mechanics and Engineering*
568 393, 114823.
- 569 Zhang, G., Y. Hu, M., Eddy Patuwo, B., C. Indro, D., 1999. Artificial neural networks in bankruptcy
570 prediction: General framework and cross-validation analysis. *European Journal of Operational*
571 *Research* 116 (1), 16–32.

572 Zhou, D., Tatomir, A., Niemi, A., Tsang, C.-F., Sauter, M., 2022. Study on the influence of randomly
573 distributed fracture aperture in a fracture network on heat production from an enhanced
574 geothermal system (EGS). Energy 250, 123781.
575

Originally published in Scientific Reports 14, article nr. 16389 (2024), Nature, July 2024, DOI 10.1038/s41598-024-64855-2.

1 Deep-learning-based Cell Segmentation for Rapid Optical Cytopathology of 2 Thyroid Cancer

3
4 *Peter R. Jermain^{1,2}, Martin Oswald³, Tenzin Langdun³, Santana Wright,¹ Ashraf Khan⁴, Thilo
5 Stadelmann,^{3,5} Ahmed Abdulkadir,^{3,6} and Anna N. Yaroslavsky^{*1,7}*

6
7 ¹Advanced Biophotonics Laboratory, University of Massachusetts Lowell, Lowell, MA, United States

8 ²Department of Radiation Medicine, MedStar Georgetown University Hospital, Washington, DC, United
9 States

10 ³Centre for Artificial Intelligence, Zurich University of Applied Sciences, Winterthur, Switzerland

11 ⁴Department of Pathology, UMASS Chan Medical School-Baystate, Springfield, MA, United States

12 ⁵ECLT European Centre for Living Technology, Venice, Italy

13 ⁶University Hospital of Old Age Psychiatry and Psychotherapy, University of Bern, Bern, Switzerland

14 ⁷ Department of Dermatology, Massachusetts General Hospital, Boston, MA United States

15 ^{*}Correspondence: Anna_Yaroslavsky@uml.edu

16
17 **Abstract:** Fluorescence polarization (Fpol) imaging of methylene blue (MB) is a promising quantitative
18 approach to thyroid cancer detection. Clinical translation of MB Fpol technology requires reduction of the
19 data analysis time that can be achieved via deep learning-based automated cell segmentation with a 2D U-
20 Net convolutional neural network. The model was trained and tested using images of pathologically diverse
21 human thyroid cells and evaluated by comparing the number of cells selected, segmented areas, and Fpol
22 values obtained using automated (AU) and manual (MA) data processing methods. Overall, the model
23 segmented 15.4% more cells than the human operator. Differences in AU and MA segmented cell areas
24 varied between -55.2% and +31.0%, whereas differences in Fpol values varied from -20.7% and +10.7%.
25 No statistically significant differences between AU and MA derived Fpol data were observed. The largest
26 differences in Fpol values correlated with greatest discrepancies in AU versus MA segmented cell areas.
27 Time required for auto-processing was reduced to ~10 seconds versus one hour required for MA data
28 processing. Implementation of the automated cell analysis makes quantitative fluorescence polarization-
29 based diagnosis clinically feasible.

30
31 **Keywords:** thyroid cancer, cytopathology, fluorescence polarization, methylene blue, automated cell
32 segmentation, semantic segmentation

34 1. Introduction

35 Thyroid cancer is the 9th most common malignancy worldwide [1]. Approximately 44,000 new cases are
36 expected to be diagnosed this year in the United States [1, 2]. Besides, studies show that over 60% of adults
37 will develop one or more benign thyroid nodules during their lifetime [3]. Benign nodule growth can be
38 influenced by various health factors (e.g., age, genetics, hormonal changes, etc.) and only about 1% will
39 result in thyroid disease [4]. Prognosis for malignant thyroid nodules largely depends on the type of cancer,

40 tumor size and stage at diagnosis, as well as subsequent medical intervention [5, 6]. Therefore, early and
41 accurate differentiation of cancerous versus non-cancerous neoplasms is imperative for guiding clinical
42 management of thyroid tumors [7, 8].

43 Fine-needle aspiration (FNA) cytopathology is an established clinical method for detecting thyroid
44 cancer [9-14]. According to the Bethesda System for Reporting Thyroid Cytopathology (TBSRTC), thyroid
45 FNAs are divided into six categories: (I) non-diagnostic, (II) benign, (III) – (V) indeterminate, which
46 include atypical, undetermined, and suspicious for malignancy specimens, and (VI) malignant [15-19].
47 Definitive assessment of cytologically indeterminate samples (TBSRTC III-V), which represent 40% of
48 thyroid FNA cases, requires histological evaluation of surgical specimens following lobectomy or complete
49 thyroidectomy [20, 21]. Surgical complications such as laryngeal nerve paralysis (5-11% of patients) or
50 hypocalcemia frequently occur (20-30%) [21-23]. In all cases where surgical resection is completed,
51 patients must be subjected to hormone replacement therapy for the remainder of their lifetime [23, 24].

52 Recently, molecular genetic testing has been implemented for the cytologically indeterminate
53 thyroid nodules [25, 26]. These assays may be able to detect malignant thyroid cells, determine the subtype
54 of thyroid cancer, and/or provide information concerning the disease prognosis by identifying specific
55 chromosomal abnormalities or gene mutations [27, 28]. While some reports have indicated that genomic
56 profiling can boost diagnostic accuracy of thyroid FNA [28], others demonstrated low specificity (25%)
57 [28, 29] and positive predictive value (35%) [29, 30]. Moreover, molecular testing requires a dedicated
58 FNA specimen that can be interpreted only in the context of a patient's complete clinical workup.
59 Widespread use of molecular testing has been hampered by substantial costs, protracted turnaround times,
60 and requirements for complex laboratory facilities and equipment [31]. Thus, development and clinical
61 implementation of accurate, quantitative, and economical methods for diagnosing thyroid neoplasms are
62 imperative for addressing cytologically indeterminate thyroid nodules.

63 Methylthioninium chloride, commonly referred to as methylene blue (MB), is a positively charged,
64 cytological stain approved for human use by the Food and Drug Administration (FDA) of the United States.
65 Studies examining the use of MB as an exogenous contrast agent have demonstrated its affinity to malignant
66 cells and tissues [32-37]. More recently, MB fluorescence polarization (Fpol) was proposed as a promising
67 tool for cancer detection [38-41]. Fluorescence polarization is an optical technique that quantifies the degree
68 of polarization of fluorescence emission relative to the polarization of incident light [42]. Whereas Fpol
69 values are sensitive to factors that limit the rotational motion of the fluorophore during the lifetime of the
70 excited state (e.g., increased local viscosity or binding), they are much less influenced by the underlying
71 optical properties of cells and tissues as compared to fluorescence emission.

72 Previously, we assessed MB Fpol of thyroid FNA specimens in a double-blind, multi-institutional
73 clinical study [43]. We demonstrated that MB Fpol is significantly higher in cancerous versus non-

cancerous cells, including specimens from the cytologically indeterminate categories. However, clinical translation of the technology is hindered by manual data processing, which makes it expensive, labor intensive and time consuming. Employing automated cell segmentation and image processing holds the potential to make Fpol cytology faster and would allow its deployment in settings where no specifically trained personnel are available. In supervised training, an artificial intelligence (AI) system can learn from examples and train an automated algorithm to process new images comparable to how a human expert would do it [44-46].

In this study we developed and evaluated a verifiable and trustworthy automated image analysis pipeline for segmenting clinically relevant cell areas and facilitating the Fpol data analysis. The goal of the project was to demonstrate feasibility of accelerating Fpol data processing, confirm diagnostic robustness of the Fpol method, and facilitate clinical translation of this promising new tool for quantitative cancer diagnosis.

2. Results

2.1 Performance of the U-Net Model

Cellular level MB Fpol values obtained by the U-Net-based automated (AU) cell segmentation are compared to manual (MA) data processing in Figure 1. The scatter plots display all cells in the test set processed with AU (Figure 1A) and MA (Figure 1B) methods. The cells are grouped into three diagnostic categories: i) malignant (papillary thyroid carcinoma, PTC; follicular thyroid carcinoma, FTC; medullary thyroid carcinoma, MTC), ii) benign (follicular thyroid adenoma, FTA; multinodular goiter, MNG), and iii) normal. Each cell is characterized by its size, calculated as region area, and the corresponding MB Fpol value. In total, 601 cells were analyzed from the test set, including 426 matched cells and 175 unmatched cells. Unmatched cells included 125 cells selected only by the model, and 50 cells chosen solely by the expert. The model selected 15.8% more cells than the human operator. Segmented areas were generally smaller with the AU method (e.g., AU area was smaller in 75% of matched cells); however, the range of Fpol values for matched and unmatched AU cells were very similar to that of the MA data. In Figure 1A (data processed by the U-Net), the range of Fpol values were 0.228 – 0.333 (malignant cells), 0.100 – 0.253 (benign cells), and 0.158 – 0.200 (normal cells). In Figure 1B (data processed manually), the ground truth cellular Fpol values ranged between 0.224 – 0.335, 0.100 – 0.226, and 0.130 – 0.204 for malignant, benign, and normal groups, respectively. In both methods cancerous cells are clustered at higher Fpol values, while non-cancerous cells (benign and normal) are distributed across lower Fpol values.

Applying the trained U-Net model to the test set and running the morphological post-processing operations and computation of Fpol values required ~10 seconds per sample on a laptop equipped with an Intel i9-12900H (Intel Corporation, Santa Clara, CA) without any need for specialized hardware such as a high-

performance graphics processor unit (GPU) or AI accelerator. In contrast, manual data processing required at least one hour to draw all the segmentation masks and perform Fpol calculations for each sample.

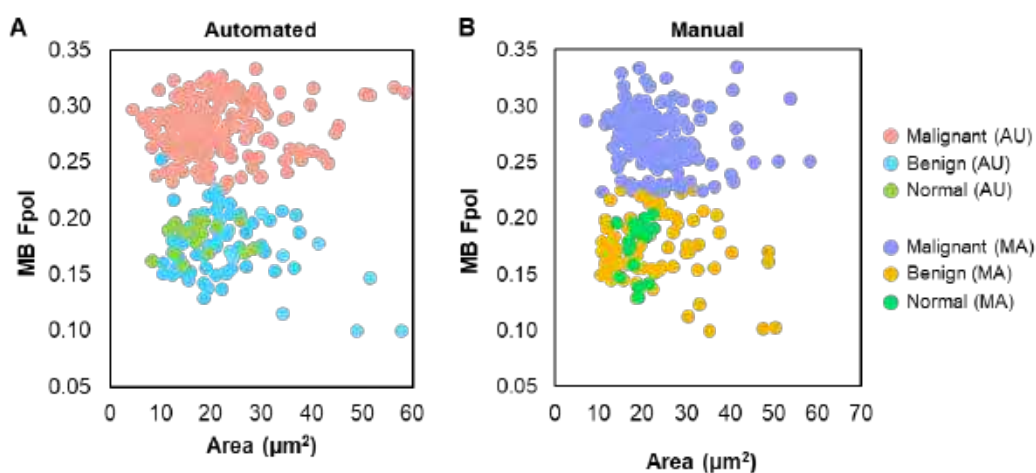


Figure 1. Scatter plot of MB Fpol value versus cell area (μm^2). Circles represent cells in the [test](#) samples processed using (A) automated and (B) manual methods.

2.2 Representative Cases

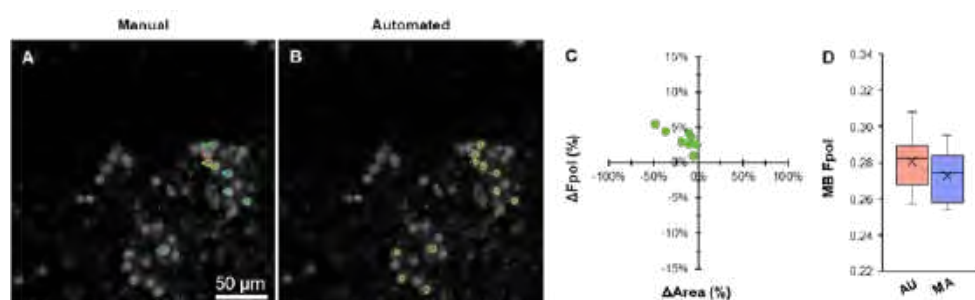
The MB Fpol technology with automated image processing may provide an accurate, clinically viable tool for differentiating cytologically indeterminate thyroid nodules as cancerous or non-cancerous, as highlighted by the following cases selected from the study [test](#) set.

2.2.1 Malignant Specimen (TBSRTC III)

Figure 2 displays a cancerous aspirate (sample #6) that yielded indeterminate results using conventional cytopathological evaluation. The nodule was categorized as TBSRTC III, and histopathology determined the definitive diagnosis as follicular variant of PTC. The MB fluorescence emission images in Figure 2A and 2B display cytomorphology, including sheets of clustered PTC cells with nuclear crowding and enlargement. Figure 2A presents cells annotated by manual segmentation (multi-color regions), while in Figure 2B the cells were segmented by the U-Net (yellow regions). Manual selection yielded 8 cells. The model segmented 11 cells, including all those selected manually, plus 3 additional unmatched cells (AU #0, #1, and #2). Detailed data for all the cells, including MB Fpol values and region areas, demonstrate good agreement between the AU and MA methods (Supplementary Table S1). Figure 2C displays a scatter plot of percent difference in Fpol versus cell area, for the matched cells selected automatically and manually. Overall, the Fpol values differed by 1.1% to 5.6%, whereas segmented areas varied between -

48.6% to -3.2%. Larger differences in Fpol correlated with greater discrepancies in cell area (e.g., in MA #6, which corresponds to AU #8, there were variations of 5.6% and -48.6% in Fpol and area, respectively). In Figure 2D, the box plot shows the distribution of Fpol values. The range of Fpol values obtained via auto-segmentation (0.257 to 0.308) was comparable to manual tracing (0.254 to 0.295). Average MB Fpol measured automatically was 0.281 ± 0.015 , whereas manually it was 0.273 ± 0.015 (difference of 2.8%).

137



138

Figure 2. Representative data for malignant PTC cells (sample #6). MB fluorescence emission image with cells (A) traced by the human expert and (B) segmented by the model. (C) Scatter plot of percent difference in MB Fpol value versus cell area (μm^2) for matched cells selected in both manual and automated methods. (D) Box plot of MB Fpol values of all cells from each method. Boxes extend from the first to the third quartiles. Black cross marks and horizontal lines represent the mean and median values, respectively. Vertical lines indicate 1.5 times the interquartile range from each box end.

145

2.2.2 Benign Specimen (TBSRTC III)

Figure 3 presents an indeterminate TBSRTC III thyroid nodule (sample #7). The final diagnosis of benign FTA was determined using histopathology. In Figure 3A and 3B, MB fluorescence emission images display clustered cells with small round nuclei and smooth nuclear margins. The same 4 cells were selected by AU and MA-based segmentation methods (Supplementary Table S2). In comparison, differences in Fpol values ranged between -8% to 10.5%, and differences in cell areas differed by -27.2% to 19.9% (Figure 3C). Larger variations between Fpol values correlated with larger deviation in cell area. For example, MA #4 and corresponding AU #3 varied by 10.5% (Fpol) and -27.2% (area). In Figure 3D, average MB Fpol values were 0.139 ± 0.039 (range: 0.100 to 0.190) and 0.137 ± 0.034 (0.103 to 0.170) for AU and MA methods, respectively (difference of 1.4%).

156

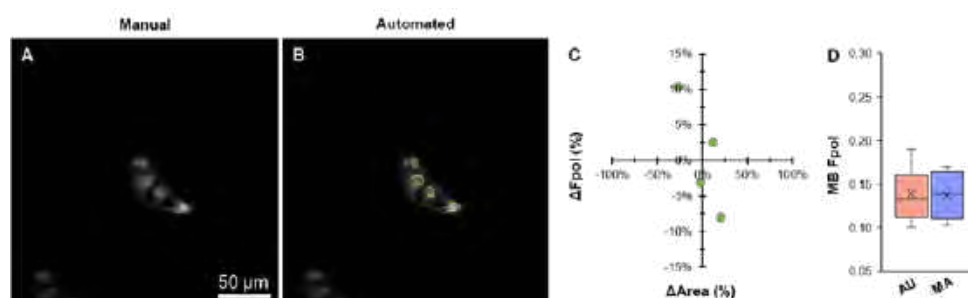


Figure 3. Representative data for benign FTA cells (sample #7). MB fluorescence emission image with cells annotated by (A) manual and (B) automated methods. (C) Percent difference in MB Fpol versus cell area (μm^2) for matched cells. (D) Distribution of MB Fpol values.

2.2.3 Malignant Specimen (TBSRTC IV)

Figure 4 shows results for a cancerous FNA (sample #5) that was categorized as TBSRTC IV. Histopathology revealed the final diagnosis as poorly differentiated FTC. The MB fluorescence emission images in Figure 4A and 4B show microfollicles with non-specific nuclear atypia, overlapping, and crowding. Manual segmentation yielded 13 cells for analysis, whereas the model segmented 16 cells, including 12 of those selected manually, plus 4 additional unmatched monomorphic cells with a round shape (AU #1, #8, #11, and #14). Detailed data for all cells are provided in Supplementary Table S3. In Figure 4C, differences in Fpol values and segmented areas varied between -3.3% to 7.8% and -9.3% to -55.2%, respectively. AU-derived Fpol values ranged from 0.276 to 0.321, while MA calculated Fpol values ranged from 0.276 to 0.325. Average MB Fpol of the samples obtained using both methods (AU: 0.301 ± 0.012 ; MA: 0.297 ± 0.014) agreed within 1.4% (Figure 4D).

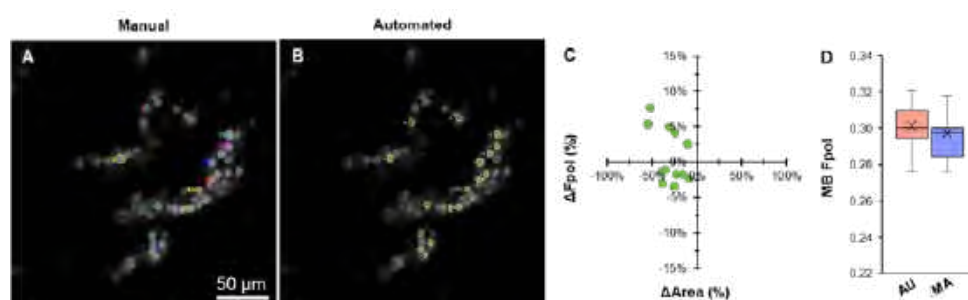


Figure 4. Representative data for malignant FTC cells (sample #5). MB fluorescence emission image with cells annotated by (A) manual and (B) automated methods. (C) Percent difference in MB Fpol versus cell area (μm^2) for matched cells. (D) Distribution of MB Fpol values.

2.3 Statistical Assessment

Figure 5 shows average MB Fpol for the test samples where cells were grouped based on data processing technique (AU or MA), diagnostic classification (malignant, benign, or normal), and/or indeterminate cytology category. In Figure 5A, the average MB Fpol value of all malignant cells (PTC, FTC, and MTC) determined by AU method was 0.279 ± 0.002 , whereas with MA technique it was 0.270 ± 0.002 (difference of 3.2%). All benign cells (FTA and MNG) had an average Fpol value of 0.174 ± 0.004 and 0.168 ± 0.004 using AU and MA methods, respectively (difference of 3.4%). In all normal cells, the AU-derived average Fpol value was 0.184 ± 0.003 , while that from the MA approach was 0.176 ± 0.005 (difference of 4.3%). Figure 5B shows average MB Fpol values for cells from the indeterminate cytology categories. In malignant cells, average Fpol values were 0.276 ± 0.002 (AU) and 0.267 ± 0.002 (MA) (difference of 3.3%), whereas for benign cells the average values were 0.176 ± 0.004 (AU) and 0.172 ± 0.003 (MA) (difference of 2.3%). Statistical analysis revealed that MB Fpol was significantly elevated in malignant versus benign ($p < 0.0001$) and malignant versus normal ($p < 0.0001$) cells in both AU and MA data sets. Importantly, there were no significant differences ($p > 0.01$) in MB Fpol values obtained via AU versus MA processing methods.

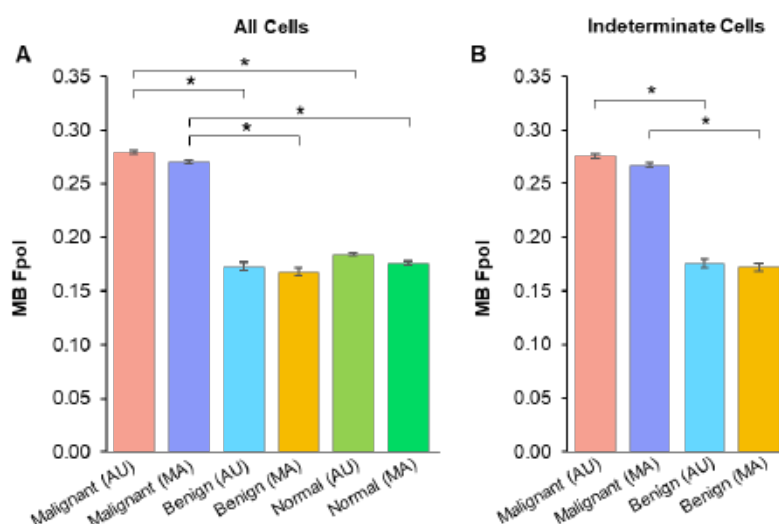


Figure 5. Average MB Fpol value of cells in the test samples processed using automated and manual methods for (A) all cells and (B) indeterminate cells. The cells were grouped by histological diagnosis: malignant (PTC, FTC, and MTC), benign (FTA and MNG), and normal. Error bars represent one standard deviation calculated over all cells in the respective category. * $p < 0.0001$.

200 3. Discussion

201 This is the first feasibility study that evaluated utility of automated segmentation for thyroid cancer
202 detection at the cellular level using Fpol imaging. Our published work has demonstrated that MB Fpol
203 provides a robust and reliable metric for quantitative differentiation of thyroid cancer in single cells [43].
204 However, the previously presented methodology required manual cell selection, segmentation, and Fpol
205 calculation by expert microscopists and pathologists. In this contribution, we introduced and evaluated the
206 U-Net-based model for rapid automated detection, segmentation, and Fpol evaluation of diagnostically
207 relevant cells in thyroid fine-needle aspirates. The results (Figure 1) demonstrate that automated patterns
208 of cell selection and segmentation agreed well with expert hand tracing performed according to
209 requirements and standards determined by pathologists. The developed model reduced time required for
210 the analysis of one sample from 1 hour to approximately 10 seconds and yielded Fpol values that are not
211 statistically different (Figure 5) from those obtained by manual assessment for all cell types and diagnoses
212 investigated.

213 On average, cell areas segmented by the U-Net were 6% smaller as compared to the manually
214 selected regions. [This trend remained unchanged after evaluating inter-operator variability on a small subset
215 of cells.](#) There are several reasons for the model to underestimate the cell extent. First, the training set
216 contoured by experts intentionally included cells with conservative boundary delineations. Due to the
217 quantitative nature of Fpol image analysis, it is imperative to avoid inclusion of background pixels that
218 could impact intensity measurements and lead to the wrong diagnostic classification. Second, as MB is a
219 nuclear stain, cell nuclei are brighter in fluorescence emission images relative to the cytoplasm and other
220 intracellular organelles. As a result, the model may preferentially select a smaller sub-region of some cells.
221 In contrast, in certain cases the model grouped multiple cells into one large region, most notably when the
222 image frame contained clustered or overlapping cells. Both these shortcomings can be addressed by
223 optimizing sample preparation, staining, and image acquisition techniques, as well as utilizing computer
224 vision methods for improving image quality, augmentation, and algorithmic modifications. [The
225 morphological operations removed spurious pixels that were segmented as cells but may have added some
226 pixels to some cells. However, because the operations were closing in nature and the segmented areas were
227 largely convex, the overall outlines of the cells did not change. Nevertheless, different parameters of the
228 morphological operations may slightly impact the segmented areas.](#) Importantly, despite these challenges,
229 AU-derived Fpol values accurately distinguished malignant thyroid cells from benign or normal (Figure 5).
230 In addition, the model segmented 15.4% more cells than the human operator. All of these cells aligned with
231 standard of care selection parameters and returned Fpol values consistent with the diagnosis of the
232 respective FNA. [Despite observed discrepancy in cell selection and systematic under-segmentation, the](#)

233 diagnostically relevant Fpol values were consistent with those from the manual segmentation. This
234 furthermore underlines the robustness of the quantitative Fpol approach.

235 The intermediate result from the AI, which is the segmentation masks, depends on the architecture,
236 the training procedure, the training data, and the post-processing such as the morphological operations.
237 With changes to some of these aspects, the overlap accuracy may increase, but the consistency in Fpol value
238 may not increase proportionally. Therefore, although there is room for improving the segmentation
239 algorithm and possibly the manual annotation, we refrained from experimenting with additional class-
240 preserving augmentations such as pixel-wise noise and elastic deformation to focus on the conceptual
241 advancement of automating extraction of Fpol values.

242 In cytopathology, the Bethesda system yields high diagnostic sensitivity/specificity (>90%) for
243 distinctly benign (TBSRTC II) or malignant (TBSRTC VI) thyroid tumors. However, diagnosing nodules
244 that fall within the indeterminate categories (TBSRTC III-V) remains a significant clinical challenge.
245 Therefore, we specifically targeted these categories when selecting the test data set. Specifically, out of
246 twelve indeterminate samples, eleven were included in the test set and one was used for training. The results
247 summarized in Figure 5B demonstrate statistically significant separation between Fpol values of malignant
248 and benign indeterminate cells obtained utilizing AU segmentation, whereas the results obtained using AU
249 and MA segmentation were not statistically significant. Quantitative results demonstrated that AU
250 evaluations can rapidly and accurately differentiate cancerous versus non-cancerous indeterminate thyroid
251 nodules (Figures 2-4). Malignant PTC and FTC cells exhibited significantly higher ($p < 0.0001$) MB Fpol
252 relative to benign FTA cells. These findings underscore specific applications where automated Fpol
253 cytology could improve upon conventional FNA assessments. For example, in the evaluation of follicular
254 thyroid lesions, FNA cannot currently distinguish FTA versus FTC due to morphological similarities
255 between the cells. Interestingly, the results presented in Figure 3 show that Fpol was capable of diagnosing
256 FTA using the images of just four cells, as compared to at least sixty cells required for traditional cytological
257 evaluation. Generally, these types of cases may be deemed suspicious for a follicular neoplasm and cannot
258 be reliably determined as malignant or benign by FNA alone. Instead, histological review of surgical
259 specimens would be needed to identify evidence of capsular invasion, vascular invasion, or invasion of the
260 adjacent parenchyma. In contrast, AU-derived Fpol measurements could reduce uncertainty and deliver a
261 prompt, accurate diagnosis directly following the FNA acquisition.

262 The AI model was trained using six samples, including three with cancerous PTC cells, which is
263 the most common subtype of thyroid cancer (80-85% of cases). Other thyroid cancers, such as FTC (10-
264 15% of cases) or MTC (4-10% of cases), were not included in the training set for this feasibility study.
265 Nonetheless, the U-Net overcame limitations and accurately processed FTC and MTC cells based on
266 morphological features learned from other training samples. While this kind of generalization is expected

from deep learning models to some extent, more pronounced differences between samples observed during training and evaluation must be accounted for with specialized domain adaptation methods [47, 48]. Automated results for FTC revealed highly elevated MB Fpol values in good agreement with the ground truth (Figure 4). There was one MTC case in the test set. Results for this aggressive thyroid cancer are shown in Figure 6. The ground truth had 18 cells (Figure 6A) and the model segmented 20 cells (Figure 6B) including 14 cells selected by the expert, plus 6 additional unmatched cells (Supplementary Table S4). The differences in Fpol values and cell areas varied from -5.7 to 13.0% and -47.8% to 26.1%, respectively (Figure 6C). Despite lack of MTC cells in the training set, average MB Fpol values (AU: 0.288 ± 0.032 ; MA: 0.277 ± 0.033) agreed within 3.8% (Figure 6D). Further improved model performance could be achieved by adding various subtypes of thyroid cancers to the training data set.

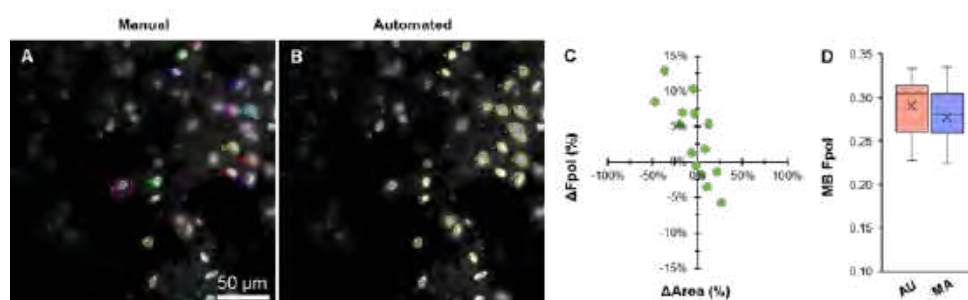


Figure 6. Representative data for malignant MTC cells (sample #10). MB fluorescence emission image with cells annotated by (A) manual and (B) automated methods. (C) Percent difference in MB Fpol versus cell area (μm^2) for matched cells. (D) Distribution of MB Fpol values.

The Fpol method offers several advantages compared to standard FNA analysis. Currently, subjective visual inspection is used to provide a diagnosis based on cytomorphological features in a process that is expensive, labor-intensive, and time-consuming. In addition, diagnostic accuracy of FNA may be impacted by the clinical expertise and experience of the reading cytopathologist. This is especially true for indeterminate cytology and borderline nodules (up to 40% of thyroid FNA cases), where atypical cytomorphology fails to detect malignancy, but prevents a straightforward benign classification. In comparison, Fpol yields conclusive results by classifying thyroid cells into malignant or benign categories based on objective and quantitative measurements. Automation accelerates the process, enabling fast cell segmentations and Fpol calculations, and delivers an accurate diagnosis within seconds. It removes the need for highly trained personnel and substantially reduces costs, creating a path for clinical implementation of automated Fpol technology.

294 In general, AI systems are not considered trustworthy for autonomous medical applications [49].
295 To improve safety while still taking full advantage of the data-driven paradigm [50], one typical approach
296 is human-AI teaming, where the AI system provides input on which a human expert can iterate or decide.
297 This is exactly the setting foreseen for Fpol technology, which can provide standalone evaluations and/or
298 augment conventional cytology in challenging cases. Once integrated into a pathology workflow, the
299 automated method provides verifiable cell segmentations and Fpol assessments that can be interpreted in
300 the context of other diagnostic exams (e.g., blood tests, etc.). *The results indicate that the model yields*
301 *considerable savings in labor and processing time as compared to the alternative manual data processing.*
302 *Automated Fpol imaging approach holds the potential to reduce the rate of indeterminate thyroid cytology,*
303 *expensive genetic testing, and unnecessary diagnostic surgeries. Future work includes development of a*
304 *fail-safe user interface for the technology, plus improved segmentation performance using a larger data set*
305 *of malignant, benign, and normal thyroid cells.*

306

307 **4. Materials and Methods**

308 *4.1 Study Design*

309 We have used data from a previously published, double-blind clinical study with excess, deidentified
310 thyroid tissues [43] to train and test a deep learning model [51-53] for rapid, quantitative detection of thyroid
311 cancer at the cellular level. The thyroid specimens were collected from malignant tumors (PTC, FTC, and
312 MTC), benign nodules (FTA and MNG), and normal thyroid epithelial tissues. *Training refers to the process*
313 *of teaching the deep learning model to segment diagnostically relevant thyroid cells based on recognizable*
314 *features and patterns in the optical images, whereas testing indicates an evaluation of the trained model*
315 *performance on tasks that included cell selection, segmentation, and Fpol data analysis.* The training
316 samples represented a range of pathologically diverse thyroid tissue types with the following diagnoses:
317 PTC, FTA, and normal thyroid cells. *Training data included six samples from four subjects that contained*
318 *90 images with 850 cells contoured by a human expert.* Test samples represented cancerous and non-
319 cancerous thyroid conditions and included the following diagnoses: PTC, FTC, MTC, FTA, MNG, and
320 normal. *Test data included seventeen samples from twelve subjects that contained 34 images with 601 cells*
321 *contoured by the model and/or human expert.* The training and test sets were non-overlapping (i.e., did not
322 include the same images or data). For each test sample, we evaluated and compared the number of cells
323 selected, segmented areas, and Fpol values obtained automatically and manually. The manually processed
324 data served as the ground truth. The results of the study were statistically analyzed using the Student's t-
325 test.

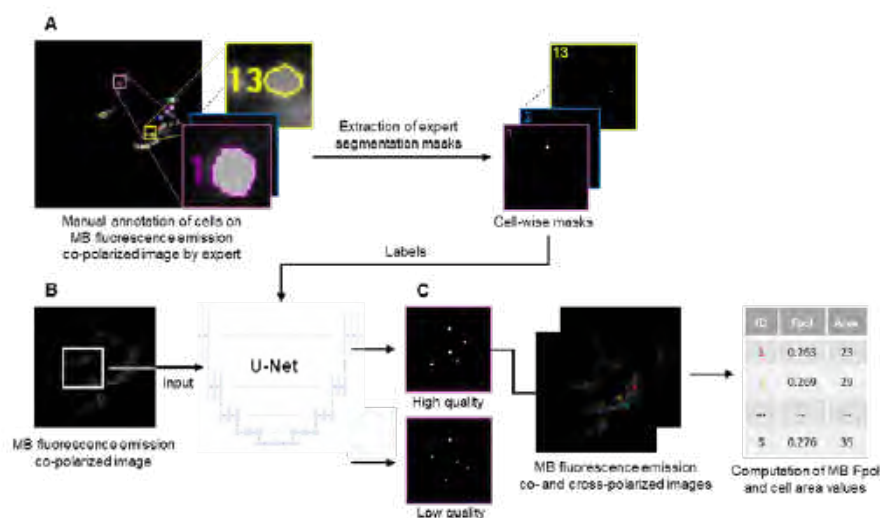
326 All methods were carried out by following relevant guidelines/regulations. The University of
327 Massachusetts Institutional Review Board (IRB) determined that this study did not constitute regulated

research as specified by the United States Food and Drug Administration (FDA) and/or Department of Health and Human Services (DHHS), as only excess, deidentified thyroid tissues obtained from surgeries were used. Therefore, the study was exempt from University of Massachusetts IRB approval, and any requirement for patient consent was waived.

4.2 Automated Segmentation Model

We used deep fully convolutional neural networks with U-Net architecture for image segmentation [52, 53], which is particularly suitable for biomedical image analysis, because of its high-resolution processing. The network is trained fully supervised end-to-end [53, 54]. Image segmentation requires a relatively small number of images for training, because each pixel constitutes an example. Previous work showed that the training annotation of the training data can even be sparse (i.e., not all pixels must have a label) [55]. In our task, the input to the network is a pair of averaged co-polarized and cross-polarized MB fluorescence emission images that corresponded to the same field of view. The in-house built, multi-modal confocal imaging system and image acquisition protocol were thoroughly described in previous publications [38, 43]. All the images were 8-bit grayscale format, with a file size of 1MB, field of view of $205\text{ }\mu\text{m}^2$ (1000×1000 pixels), lateral resolution better than $0.9\text{ }\mu\text{m}$, and axial resolution of $2\text{ }\mu\text{m}$. The images were processed with a relatively standard U-Net with five resolution levels (architectural details and training procedure are described in Supplementary Methods). The output produced two segmentation masks: one for cells with high quality signal and another for cells with low quality. Low quality of individual cells was observed, for example, when they were out of focus. The algorithm computed MB Fpol and cell area values for each high quality cell.

350



350

Figure 7. Outline of the segmentation model. (A) Ground truth labels of cells based on expert manual annotation were converted into segmentation masks for the AI training. (B) Masks and MB fluorescence emission images enter supervised training of the U-Net as labels and input, respectively. (C) Algorithm segments the cells and computes the corresponding MB Fpol and cell area values.

4.3 Model Training

The cytological samples that were set aside as a test set were not used during the training of the network to avoid using the same data once in training and then again during testing [52, 53]. The U-Net was trained with a two-channel input (averaged co-polarized and cross-polarized MB fluorescence emission images) and corresponding label masks for diagnostically relevant cells in the images. All label masks in the training set were generated based on annotation masks produced by an expert in optical microscopy and verified by a study pathologist. Figure 8 presents example data for the U-Net. A MB co-polarized fluorescence emission image, utilized as input to the deep neural network, is shown in Figure 8A. It displays cytomorphological features of the cells (e.g., cell membrane, nuclear size and shape, etc.). Figures 8B–D show binary masks of the background (Figure 8B), regions labeled as cell-like structures (Figure 8C), and regions labelled as high quality cells (Figure 8D). The masks of high-quality cells were drawn by an expert trained in microscopy and pathology, and selected because they were deemed diagnostically relevant. However, the expert did not outline all diagnostically relevant cells, meaning that a lack of a mask over a cell does not imply that the cell is not diagnostically relevant. Furthermore, the distinction between diagnostically relevant and irrelevant cells is not definitive, leading to ambiguity in the labels. We therefore introduced one output for the high-quality cells and one for ambiguous cells. This reduces negative effects of false positive predictions of high-quality cells that resemble ambiguous cells. The cross-entropy loss with equal weight for all outputs was optimized using adaptive moment estimation and batch size of four. We applied online augmentations to reduce overfitting, including rotation, flips, scaling, shearing, elastic transformations, and random cropping.

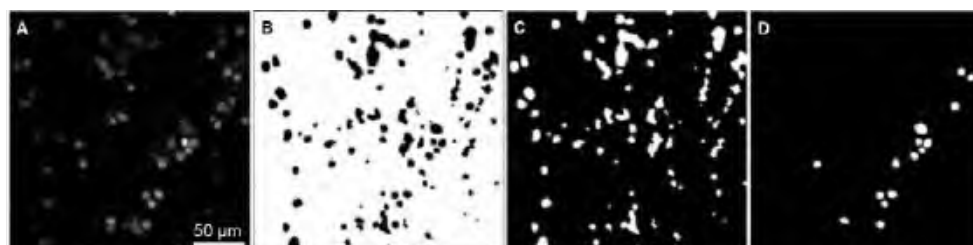


Figure 8. Example data for the network. (A) Input co-polarized MB fluorescence emission image. It is an 8-bit, grayscale image (1000×1000 pixels). Binary masks of (B) background, (C) ambiguous cell signal, and (D) high-quality cells.

Model training and testing was performed on a system equipped with an NVIDIA T4 GPU (NVIDIA, Santa Clara, CA) with 16GB of video RAM, 70 watts of thermal design power, and theoretical 8.1 TFLOPS of peak computing power. The training required up to 8 hours. Additional training time would be needed without dedicated hardware, whereas the time would be shorter if a more powerful GPU were used.

4.4 Post-Processing

Following the segmentation, and prior to the quantitative analysis, we improved the sensitivity and specificity of the binary segmentation masks with post-processing operations. The pixel-wise probability for foreground and background was binarized with a threshold of 0.5. We corrected some segmentation inconsistencies and increased image clarity through a binary morphological process. We applied a dilation operation with a 3×3 kernel in three iterations, which expands segmented regions, thereby resolving small gaps in the objects. Following this, an erosion operation, mirroring the kernel size and iteration parameters of dilation, was performed to revert the dilated areas to their initial boundaries. We eliminated some spurious clusters of pixels identified as cells with a morphological opening operation using an elliptical structuring element with a 15×15 kernel size, which offers versatility to accommodate diverse noise shapes.

4.5 Fpol Calculations

In the binary segmentation mask, individual high quality cells were defined and identified via connected component labeling. Within each cell, the average Fpol value was computed. Averaged co-polarized and cross-polarized MB fluorescence emission images were generated, then a threshold was applied to eliminate saturated pixels and background (pixel values > 254 and < 3, respectively). Mean pixel values across the cell area were used to calculate MB Fpol:

$$F_{pol} = (I_{co} - G \cdot I_{cross}) / (I_{co} + G \cdot I_{cross}) \quad (1)$$

where I_{co} and I_{cross} represent averaged co-polarized and cross-polarized fluorescence intensities, respectively. $G = 0.75$ is the calibration factor of the microscopy system [43].

4.6 Performance Evaluation

For the quantitative evaluation, we analyzed the *test* set using AU and MA data processing methods. The MA results were obtained previously and represented the ground truth [43]. The *test* samples were processed using the U-Net, then segmented cells were enumerated via connected component labeling and Fpol values were auto-calculated – as described above in subsection 2.5. For each *test* case, we compared cell selections, total number of selected cells, segmented region areas, and Fpol values obtained automatically with the corresponding parameters from the ground truth. Cells segmented by both AU and MA methods are referred to as *matched* cells. However, some AU selected cells may have no corresponding MA selected cell and vice-versa (i.e., *unmatched* cells). Percent differences in Fpol values or cell areas between AU and MA defined matched cells were calculated as follows:

$$\Delta Fpol = \frac{Fpol_1 - Fpol_2}{\left(\frac{Fpol_1 + Fpol_2}{2}\right)} \times 100\% \quad (2)$$

$$\Delta Area = \frac{Area_1 - Area_2}{\left(\frac{Area_1 + Area_2}{2}\right)} \times 100\% \quad (3)$$

where $\Delta Fpol$ and $\Delta Area$ represent the percent differences, and $Fpol_1$ and $Fpol_2$, and $Area_1$ and $Area_2$, are the respective quantities obtained for the cell via AU and MA methods, respectively.

Due to different segmentation protocols, cells in the AU and MA data sets may have different region labels (e.g., automatically selected cell AU #1 may correspond to manually selected cell MA #2, etc.).

4.7 Statistical Analysis

Statistical analysis was performed to evaluate the significance of differences between the AU and MA processing techniques [56, 57]. It tested the null hypothesis that there are significant differences between Fpol values of malignant versus benign cells and malignant versus normal cells, and that there are no significant differences in MB Fpol values derived from AU versus MA processing methods. The cells were organized into six groups: *i*) malignant (AU processing), *ii*) benign (AU processing), *iii*) normal (AU processing), *iv*) malignant (MA processing), *v*) benign (MA processing), and *vi*) normal (MA processing). In addition, eleven indeterminate samples were analyzed separately. They were organized in 4 groups: *i*) malignant (AU processing), *ii*) benign (AU processing), *iii*) malignant (MA processing), and *iv*) benign (MA processing). Average Fpol values and corresponding standard deviations were obtained for each group. The distribution of the Fpol values across the samples was confirmed to be normal, therefore the paired two-tailed Student's t-tests were utilized. The significance was Bonferroni corrected [57]. The differences were considered statistically significant when $p < 0.01$.

References

1. Cancer Facts & Figures, 2023. The American Cancer Society.
<https://www.cancer.org/research/cancer-facts-statistics/all-cancer-facts-figures/2023-cancer-facts-figures.pdf> (accessed on September 23, 2023).
2. Vaccarella S, Franceschi S, Bray F, Wild CP, Plummer M, Dal Maso L. Worldwide Thyroid-Cancer Epidemic? The Increasing Impact of Overdiagnosis. *N Engl J Med*. 2016 Aug 18;375(7):614-7. doi: 10.1056/NEJMp1604412. PMID: 27532827.
3. Haymart MR. Progress and Challenges in Thyroid Cancer Management. *Endocr Pract*. 2021 Dec;27(12):1260-1263. doi: 10.1016/j.eprac.2021.09.006. Epub 2021 Sep 22. PMID: 34562612.
4. Cleveland Clinic. Thyroid Cancer: Types, Symptoms, Causes & Treatment. <https://my.clevelandclinic.org/health/diseases/12210-thyroid-cancer> (Accessed Dec 19, 2023)
5. Davies L, Welch HG. Current thyroid cancer trends in the United States. *JAMA Otolaryngol Head Neck Surg*. 2014 Apr;140(4):317-22. doi: 10.1001/jamaoto.2014.1. PMID: 24557566.
6. Uppal N, Collins R, James B. Thyroid nodules: Global, economic, and personal burdens. *Front Endocrinol (Lausanne)*. 2023 Jan 23;14:1113977. doi: 10.3389/fendo.2023.1113977. PMID: 36755911; PMCID: PMC9899850.
7. Sira L, Balogh Z, Vitális E, Kovács D, Győry F, Molnár C, Bodor M, Nagy EV. Case Report: Medullary Thyroid Cancer Workup Initiated by Unexpectedly High Procalcitonin Level-Endocrine Training Saves Life in the COVID-19 Unit. *Front Endocrinol (Lausanne)*. 2021 Oct 11;12:727320. doi: 10.3389/fendo.2021.727320. PMID: 34707568; PMCID: PMC8543013.
8. Isik A, Firat D, Yilmaz I, Peker K, Idiz O, Yilmaz B, Demiryilmaz I, Celebi F. A survey of current approaches to thyroid nodules and thyroid operations. *Int J Surg*. 2018 Jun;54(Pt A):100-104. doi: 10.1016/j.ijsu.2018.04.037. Epub 2018 Apr 28. PMID: 29709542.
9. Bibbo M. Thyroid fine needle aspiration. *Acta Cytol*. 2009 Sep-Oct;53(5):489-90. doi: 10.1159/000325373. PMID: 19798874.
10. Feldkamp J, Führer D, Luster M, Musholt TJ, Spitzweg C, Schott M. Fine Needle Aspiration in the Investigation of Thyroid Nodules. *Dtsch Arztebl Int*. 2016 May 20;113(20):353-9. doi: 10.3238/arztebl.2016.0353. PMID: 27294815; PMCID: PMC4906830.
11. Willms A, Melder J, Hoffmann MA, Wieler HJ, Prechtel H, Weitzel C, Sattler B, Trampert L, Willms D, Schwab R. The significance of FNAC in diagnosing differentiated thyroid cancer and the discrepancy between theory and practice - a multi-centre study. *Nuklearmedizin*. 2020 Jun;59(3):248-255. English. doi: 10.1055/a-1105-6200. Epub 2020 Feb 26. PMID: 32102094.
12. Howitt BE, Chang S, Eszlinger M, Paschke R, Drage MG, Krane JF, Barletta JA. Fine-needle aspiration diagnoses of noninvasive follicular variant of papillary thyroid carcinoma. *Am J Clin Pathol*. 2015 Dec;144(6):850-7. doi: 10.1309/AJCPEIE12POICULI. PMID: 26572991.
13. Misiakos EP, Margari N, Meristoudis C, Machairas N, Schizas D, Petropoulos K, Spathis A, Karakitsos P, Machairas A. Cytopathologic diagnosis of fine needle aspiration biopsies of thyroid nodules. *World J Clin Cases*. 2016 Feb 16;4(2):38-48. doi: 10.12998/wjcc.v4.i2.38. PMID: 26881190; PMCID: PMC4733475.
14. Alshaikh S, Harb Z, Aljufairi E, Almahari SA. Classification of thyroid fine-needle aspiration cytology into Bethesda categories: An institutional experience and review of the literature. *Cytojournal*. 2018 Feb 16;15:4. doi: 10.4103/cytojournal.cytojournal_32_17. PMID: 29531571; PMCID: PMC5841007.
15. Bongiovanni M, Spitale A, Faquin WC, Mazzucchelli L, Baloch ZW. The Bethesda System for Reporting Thyroid Cytopathology: a meta-analysis. *Acta Cytol*. 2012;56(4):333-9. doi: 10.1159/000339959. Epub 2012 Jul 25. PMID: 22846422.
16. Cibas ES, Ali SZ. The 2017 Bethesda System for Reporting Thyroid Cytopathology. *Thyroid*. 2017 Nov;27(11):1341-1346. doi: 10.1089/thy.2017.0500. PMID: 29091573.
17. Naz S, Hashmi AA, Khurshid A, Faridi N, Edhi MM, Kamal A, Khan M. Diagnostic accuracy of Bethesda system for reporting thyroid cytopathology: an institutional perspective. *Int Arch Med*. 2014 Oct 31;7:46. doi: 10.1186/1755-7682-7-46. PMID: 25945126; PMCID: PMC4413982.

- 495 18. Rai K, Park J, Gokhale S, Irshaidat F, Singh G. Diagnostic Accuracy of the Bethesda System for
496 Reporting Thyroid Cytopathology (TBSRTC): An Institution Experience. *Int J Endocrinol*. 2023 Nov
497 15;2023:9615294. doi: 10.1155/2023/9615294. PMID: 38021084; PMCID: PMC10665099.
- 498 19. Mahajan S, Manjula BV, Vaishnavi R, John N, Babu B. Usefulness of The Bethesda System of
499 Reporting Thyroid Cytopathology in Surgical Planning. *Indian J Otolaryngol Head Neck Surg*. 2022
500 Oct;74(Suppl 2):2623-2628. doi: 10.1007/s12070-020-02335-5. Epub 2021 Jan 7. PMID: 36452664;
501 PMCID: PMC9702001.
- 502 20. Christou N, Mathonnet M. Complications after total thyroidectomy. *J Visc Surg*. 2013 Sep;150(4):249-
503 56. doi: 10.1016/j.jviscsurg.2013.04.003. Epub 2013 Jun 6. PMID: 23746996.
- 504 21. Vuong HG, Ngo HTT, Bychkov A, Jung CK, Vu TH, Lu KB, Kakudo K, Kondo T. Differences in
505 surgical resection rate and risk of malignancy in thyroid cytopathology practice between Western and
506 Asian countries: A systematic review and meta-analysis. *Cancer Cytopathol*. 2020 Apr;128(4):238-
507 249. doi: 10.1002/cncy.22228. Epub 2019 Dec 28. PMID: 31883438.
- 508 22. Giordano D, Botti C, Piana S, Zini M, Frasoldati A, Luseti F, Cavuto S, Savoldi L, Pernice C, Ghidini
509 A. Postoperative hypoparathyroidism after completion thyroidectomy for well-differentiated thyroid
510 cancer. *Eur J Endocrinol*. 2021 Aug 3;185(3):413-419. doi: 10.1530/EJE-21-0353. PMID: 34232122.
- 511 23. Schumm MA, Lechner MG, Shu ML, Ochoa JE, Kim J, Tseng CH, Leung AM, Yeh MW. Frequency
512 of Thyroid Hormone Replacement After Lobectomy for Differentiated Thyroid Cancer. *Endocr Pract*.
513 2021 Jul;27(7):691-697. doi: 10.1016/j.eprac.2021.01.004. Epub 2021 Jan 15. PMID: 33642257.
- 514 24. Hannoush ZC, Weiss RE. Thyroid Hormone Replacement in Patients Following Thyroidectomy for
515 Thyroid Cancer. *Rambam Maimonides Med J*. 2016 Jan;7(1):e0002. doi: 10.5041/RMMJ.10229.
516 PMID: 26886951; PMCID: PMC4737508.
- 517 25. Zhang M, Lin O. Molecular Testing of Thyroid Nodules: A Review of Current Available Tests for
518 Fine-Needle Aspiration Specimens. *Arch Pathol Lab Med*. 2016 Dec;140(12):1338-1344. doi:
519 10.5858/arpa.2016-0100-RA. Epub 2016 Aug 24. PMID: 27557410.
- 520 26. Vargas-Salas S, Martínez JR, Urra S, Domínguez JM, Mena N, Uslar T, Lagos M, Henríquez M,
521 González HE. Genetic testing for indeterminate thyroid cytology: review and meta-analysis. *Endocr*
522 *Relat Cancer*. 2018 Mar;25(3):R163-R177. doi: 10.1530/ERC-17-0405. Epub 2017 Dec 18. PMID:
523 29255094; PMCID: PMC5799829.
- 524 27. Campenni A, Siracusa M, Ruggeri RM, Laudicella R, Pignata SA, Baldari S, Giovannella L.
525 Differentiating malignant from benign thyroid nodules with indeterminate cytology by ^{99m}Tc-MIBI
526 scan: a new quantitative method for improving diagnostic accuracy. *Sci Rep*. 2017 Jul 21;7(1):6147.
527 doi: 10.1038/s41598-017-06603-3. PMID: 28733644; PMCID: PMC5522471.
- 528 28. Steward DL, Kloos RT. Clinical diagnostic gene expression thyroid testing. *Otolaryngol Clin North*
529 *Am*. 2014 Aug;47(4):573-93. doi: 10.1016/j.otc.2014.04.009. PMID: 25041959.
- 530 29. Sipos JA, Ringel MD. Molecular testing in thyroid cancer diagnosis and management. *Best Pract Res*
531 *Clin Endocrinol Metab*. 2023 Jan;37(1):101680. doi: 10.1016/j.beem.2022.101680. Epub 2022 May
532 28. PMID: 35691860.
- 533 30. Daniels GH. What is the role of molecular markers in the management of "indeterminate" thyroid
534 nodules? *Cancer Cytopathol*. 2013 May;121(5):223-4. doi: 10.1002/cncy.21289. PMID: 23670834.
- 535 31. Marcadis AR, Valderrabano P, Ho AS, Tepe J, Swartzwelder CE, Byrd S, Sacks WL, Untch BR, Shaha
536 AR, Xu B, Lin O, Ghossein RA, Wong RJ, Marti JL, Morris LGT. Interinstitutional variation in
537 predictive value of the ThyroSeq v2 genomic classifier for cytologically indeterminate thyroid nodules.
538 *Surgery*. 2019 Jan;165(1):17-24. doi: 10.1016/j.surg.2018.04.062. Epub 2018 Oct 22. PMID:
539 30360906; PMCID: PMC6289715.
- 540 32. Snuderl M, Wirth D, Sheth SA, Bourne SK, Kwon CS, Ancukiewicz M, Curry WT, Frosch MP,
541 Yaroslavsky AN. Dye-enhanced multimodal confocal imaging as a novel approach to intraoperative
542 diagnosis of brain tumors. *Brain Pathol*. 2013 Jan;23(1):73-81. doi: 10.1111/j.1750-
543 3639.2012.00626.x. Epub 2012 Aug 28. PMID: 22882328; PMCID: PMC8028953.

- 544 33. Wirth D, Snuderl M, Sheth S, Kwon CS, Frosch MP, Curry W, Yaroslavsky AN. Identifying brain
545 neoplasms using dye-enhanced multimodal confocal imaging. *J Biomed Opt.* 2012 Feb;17(2):026012.
546 doi: 10.1117/1.JBO.17.2.026012. PMID: 22463044.
- 547 34. Gill WB, Huffman JL, Lyon ES, Bagley DH, Schoenberg HW, Straus FH 2nd. Selective surface
548 staining of bladder tumors by intravesical methylene blue with enhanced endoscopic identification.
549 *Cancer.* 1984 Jun 15;53(12):2724-7. doi: 10.1002/1097-0142(19840615)53:12<2724::aid-
550 cncr2820531230>3.0.co;2-b. PMID: 6202385.
- 551 35. Kaisary AV, Smith PJ. Prazosin, priapism and management. *Br J Urol.* 1986 Apr;58(2):227-8. doi:
552 10.1111/j.1464-410x.1986.tb09036.x. PMID: 3697645.
- 553 36. Fedorak IJ, Ko TC, Gordon D, Flisak M, Prinz RA. Localization of islet cell tumors of the pancreas: a
554 review of current techniques. *Surgery.* 1993 Mar;113(3):242-9. PMID: 8382841.
- 555 37. Hillary SL, Guillermet S, Brown NJ, Balasubramanian SP. Use of methylene blue and near-infrared
556 fluorescence in thyroid and parathyroid surgery. *Langenbecks Arch Surg.* 2018 Feb;403(1):111-118.
557 doi: 10.1007/s00423-017-1641-2. Epub 2017 Dec 11. PMID: 29230539; PMCID: PMC5805804.
- 558 38. Yaroslavsky AN, Feng X, Muzikansky A, Hamblin MR. Fluorescence Polarization of Methylene Blue
559 as a Quantitative Marker of Breast Cancer at the Cellular Level. *Sci Rep.* 2019 Jan 30;9(1):940. doi:
560 10.1038/s41598-018-38265-0. PMID: 30700827; PMCID: PMC6353996.
- 561 39. Feng X, Muzikansky A, Ross AH, Hamblin MR, Jermain PR, Yaroslavsky AN. Multimodal
562 quantitative imaging of brain cancer in cultured cells. *Biomed Opt Express.* 2019 Jul 29;10(8):4237-
563 4248. doi: 10.1364/BOE.10.004237. PMID: 31453007; PMCID: PMC6701554.
- 564 40. Malik S, Jermain PR, Feng X, Yaroslavsky AN. Multimodal optical imaging of renal cells. *Opt. Eng.*
565 2019 Aug; 58, 082415. doi: 10.1117/1.OE.58.8.082415
- 566 41. Mitrou A, Yaroslavsky AN. Exogenous fluorescence polarization imaging for cancer detection in
567 optical Spectroscopy and Imaging for Cancer Diagnostics: Fundamentals, Progress, And Challenges.
568 2023 Feb; pp 139-180. doi:10.1142/9789811258961_0006
- 569 42. Jameson DM, Ross JA. Fluorescence polarization/anisotropy in diagnostics and imaging. *Chem Rev.*
570 2010 May 12;110(5):2685-708. doi: 10.1021/cr900267p. PMID: 20232898; PMCID: PMC2868933.
- 571 43. Jermain PR, Fischer AH, Joseph L, Muzikansky A, Yaroslavsky AN. Fluorescence Polarization
572 Imaging of Methylene Blue Facilitates Quantitative Detection of Thyroid Cancer in Single Cells.
573 *Cancers (Basel).* 2022 Mar 5;14(5):1339. doi: 10.3390/cancers14051339. PMID: 35267647; PMCID:
574 PMC8908998.
- 575 44. Schmidhuber J. Deep learning in neural networks: An overview. *Neural networks.* 2015 Jan 1;61:85-
576 117. doi: /10.1145/3351095.3372834
- 577 45. Suganyadevi S, Seethalakshmi V, Balasamy K. A review on deep learning in medical image analysis.
578 *Int J Multimed Inf Retr.* 2022;11(1):19-38. doi:10.1007/s13735-021-00218-1
- 579 46. Nazir S, Dickson DM, Akram MU. Survey of explainable artificial intelligence techniques for
580 biomedical imaging with deep neural networks. *Comput Biol Med.* 2023 Apr;156:106668. doi:
581 10.1016/j.compbiomed.2023.106668. Epub 2023 Feb 18. PMID: 36863192.
- 582 47. Amirian M, Montoya-Zegarra JA, Gruss J, Stebler YD, Bozkir AS, Calandri M, Schwenker FF,
583 Stadelmann T. PrepNet: A Convolutional Auto-Encoder to Homogenize CT Scans for Cross-Dataset
584 Medical Image Analysis. 2021 14th International Congress on Image and Signal Processing,
585 BioMedical Engineering and Informatics (CISP-BMEI), Shanghai, China, 2021, pp. 1-7, doi:
586 10.1109/CISP-BMEI53629.2021.9624344.
- 587 48. Sager P, Salzmann S, Burn F, Stadelmann T. Unsupervised Domain Adaptation for Vertebrae
588 Detection and Identification in 3D CT Volumes Using a Domain Sanity Loss. *J Imaging.* 2022 Aug
589 19;8(8):222. doi: 10.3390/jimaging8080222. PMID: 36005465; PMCID: PMC9410021.
- 590 49. Toreini E, Aitken M, Coopamootoo K., Elliott K., Gonzalez Zelaya C., van Moorsel A. The
591 relationship between trust in AI and Trustworthy Machine Learning Technologies. Proceedings of the
592 2020 conference on fairness, accountability, and transparency. 2020. Available from:
593 <https://doi.org/10.1145/3351095.3372834>

- 594 50. Stadelmann, T., Klamt, T., & Merkt, P. H. (2022). Data centrism and the core of Data Science as a
 595 scientific discipline. Archives of Data Science, Series A, 8(2).
 596 51. Ronneberger O, Fischer P, Brox T. U-Net: Convolutional Networks for Biomedical Image
 597 Segmentation [Internet]. 2015 [cited 2024 Jan 17]. Available from: <https://arxiv.org/abs/1505.04597>
 598 52. Deep Learning (Goodfellow IJ, Bengio Y, and Courville A), MIT Press, 2016.
 599 53. Deep Learning Foundations and Concepts (Bishop CM and Bishop H), Springer, 2024.
 600 54. Artificial Neural Networks (Gacovski Z), Arcler Press LLC, 2017
 601 55. Çiçek Ö, Abdulkadir A, Lienkamp, SS, Brox, T, Ronneberger, O (2016). 3D U-Net: Learning Dense
 602 Volumetric Segmentation from Sparse Annotation. In: Ourselin, S., Joskowicz, L., Sabuncu, M., Unal,
 603 G., Wells, W. (eds) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016.
 604 MICCAI 2016. Lecture Notes in Computer Science(), vol 9901. Springer, Cham.
 605 https://doi.org/10.1007/978-3-319-46723-8_49
 606 56. Medical Biostatistics (Indrayan A, Malhotra RK), Chapman & Hall/CRC Biostatistics Series, 4th
 607 Edition, 2017
 608 57. Medical Statistics: A Textbook for the Health Sciences (Walters SJ, Campbell MJ, Machin D), Wiley,
 609 5th Edition, 2021

610
 611
 612 **Author Contributions:** Conceptualization, A.N.Y., A.K., P.R.J., T.S., and A.A.; Methodology, A.N.Y.,
 613 A.K., P.R.J., T.S., A.A., M.O., and T.L.; Validation, P.R.J., M.O., S.W., A.A., A.K., A.N.Y., and T.S.;
 614 Formal Analysis, P.R.J., A.N.Y., S.W., A.A., M.O., A.K., T.L., and T.S.; Investigation, P.R.J., A.N.Y.,
 615 M.O., S.W., A.A., A.K., T.S., and T.L.; Resources, A.N.Y., A.K., and T.S.; Data Curation, P.R.J., A.K.,
 616 M.O., A.A., S.W., A.N.Y., and T.S.; Writing – Original Draft Preparation, P.R.J., A.N.Y., A.K., A.A.,
 617 S.W., M.O., and T.S.; Writing – Review & Editing, P.R.J., A.N.Y., A.K., A.A., S.W., M.O., and T.S.;
 618 Visualization, P.R.J., A.N.Y., A.A., M.O., S.W., and T.S.; Supervision, A.N.Y., and T.S.; Project
 619 Administration, A.N.Y. and T.S.; Funding Acquisition, A.N.Y. and T.S.

620 621 **Patents**

622 Yaroslavsky AN, Patel R, Wirth D. “Devices and Methods for Optical Pathology.” PCT/US2012/025678
 623

624 **Data Availability Statement:** The experimental data, reference model, and code are available from the
 625 corresponding author upon request.
 626

627 **Competing interests Statement:** Authors declare that they have no competing interests.
 628

629 **Funding:** This research was funded by the Massachusetts Acorn Innovation Fund and the University of
 630 Massachusetts Medical Device Innovation Program – Center for Clinical and Translational Sciences.
 631

632 **Institutional Review Board Statement:** Not applicable. The Institutional Review Board of University of
 633 Massachusetts has determined that this study was not regulated research as defined by DHHS (Department

of Health and Human Services) and FDA (Food and Drug Administration) regulations, because only unidentified, excess thyroid tissues from surgeries were used. Therefore, this study was not regulated by the ethic approval committee.

Informed Consent Statement: Not applicable. The requirement for patient consent was waived because only unidentified, excess thyroid tissues from surgeries were used.

Legends:

Figure 1. Scatter plot of MB Fpol value versus cell area (μm^2). Circles represent cells in the **test** samples processed using (A) automated and (B) manual methods.

Figure 2. Representative data for malignant PTC cells (sample #6). MB fluorescence emission image with cells (A) traced by the human expert and (B) segmented by the model. (C) Scatter plot of percent difference in MB Fpol value versus cell area (μm^2) for matched cells selected in both manual and automated methods. (D) Box plot of MB Fpol values of all cells from each method. Boxes extend from the first to the third quartiles. Black cross marks and horizontal lines represent the mean and median values, respectively. Vertical lines indicate 1.5 times the interquartile range from each box end.

Figure 3. Representative data for benign FTA cells (sample #7). MB fluorescence emission image with cells annotated by (A) manual and (B) automated methods. (C) Percent difference in MB Fpol versus cell area (μm^2) for matched cells. (D) Distribution of MB Fpol values.

Figure 4. Representative data for malignant FTC cells (sample #5). MB fluorescence emission image with cells annotated by (A) manual and (B) automated methods. (C) Percent difference in MB Fpol versus cell area (μm^2) for matched cells. (D) Distribution of MB Fpol values.

Figure 5. Average MB Fpol value of cells in the **test** samples processed using automated and manual methods for (A) **all cells** and (B) **indeterminate cells**. The cells were grouped by histological diagnosis: malignant (PTC, FTC, and MTC), benign (FTA and MNG), and normal. Error bars represent one standard deviation calculated over all cells in the respective category. * $p < 0.0001$.

667 **Figure 6.** Representative data for malignant MTC cells (sample #10). MB fluorescence emission image
668 with cells annotated by (A) manual and (B) automated methods. (C) Percent difference in MB Fpol versus
669 cell area (μm^2) for matched cells. (D) Distribution of MB Fpol values.

670

671 **Figure 7.** Outline of the segmentation model. (A) Ground truth labels of cells based on expert manual
672 annotation were converted into segmentation masks for the AI training. (B) Masks and MB fluorescence
673 emission images enter supervised training of the U-Net as labels and input, respectively. (C) Algorithm
674 segments the cells and computes the corresponding MB Fpol and cell area values.

675

676 **Figure 8.** Example data for the network. (A) Input co-polarized MB fluorescence emission image. It is an
677 8-bit, grayscale image (1000×1000 pixels). Binary masks of (B) background, (C) ambiguous cell signal,
678 and (D) high-quality cells.

Originally published in Proceedings of the Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition and Structural and Syntactic Pattern Recognition (S+SSPR'24), 2024.

Community-Hop: Enhancing Node Classification through Community Preference

Ahmed Begga¹, Waqar Ali², Gabriel Niculescu¹, Francisco Escolano¹, Thilo Stadelmann^{3,4}, and Marcello Pelillo^{2,3}

¹ University of Alicante, Alicante, Spain

² Ca' Foscari University of Venice, Venice, Italy

³ ECLT European Centre for Living Technology, Venice, Italy

⁴ ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland

Abstract. In recent years, Graph Neural Networks (GNNs) have demonstrated significant influence on the analysis of graph structures by leveraging message-passing mechanisms to aggregate neighborhood information and perform various graph-related tasks from node classification to link prediction. Recently, GNNs have mostly been developed to deal with different types of graph structures, such as homophily (similar labels among connected nodes) and heterophily (dissimilar labels among connected nodes). However, existing methods lack the ability to combine node features and graph topology optimally to deal with heterophily. This paper proposes a Community-HOP-based GNN model for dealing with homophilic and heterophilic graph structures. Specifically, we incorporate valuable insights from the graph community structure to guide the feature aggregation process of the GNN layer to learn diverse graph properties and improve performance on node-level tasks. Extensive experiments on six node-level datasets under standard metrics demonstrate that the Community-HOP method surpasses existing baselines.

Keywords: Graph Neural Networks · Node Classification · Spectral Clustering.

1 Introduction

GNNs have proven to be powerful methods for analyzing graph-based data, finding use in diverse areas such as social network analysis and predicting molecular properties [9,18,8]. However, conventional GNN architectures often face challenges in capturing complex structural information and relationships between nodes at various distances, particularly in graphs with heterophilic properties (where connected nodes have dissimilar labels) [21]. Recent advancements in GNN design have addressed these limitations by incorporating higher-order neighborhood information. Models such as MixHop [1] and FSGNN [12] have demonstrated the effectiveness of aggregating features from nodes at different hop distances. These approaches allow for more flexible feature mixing and improved performance on various graph datasets.

2 Begga et al.

While these methods have shown promise, they often treat all neighbors equally within each hop distance, potentially overlooking important structural information encoded in the graph's community structure. Community detection in graphs has long been a subject of study in network science [15,11], with spectral methods providing powerful tools for identifying clusters of densely connected nodes [3].

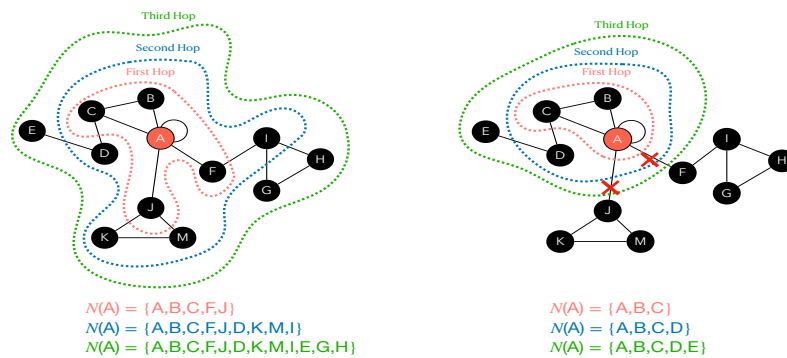


Fig. 1: Traditional Hop vs Community Hop. Evolution of the neighborhood of A, $N(A)$, in traditional hops (left) and in our approach, Community Hop (right).

This study develops a novel GNN layer that leverages graph community structure to guide the feature aggregation process. By combining spectral community detection techniques [11] with a modified transition matrix for inter-community hops, our approach aims to prioritize information flow within and between communities in a more meaningful way. This community-aware feature aggregation strategy allows the model to capture both local (by just combining community nodes) and global (by using the same GNN for all the clusters) graph structures more effectively.

2 Related Work

Recent years have seen significant advancements in GNN architectures, particularly in addressing the challenges of heterophily and over-smoothing. These innovations have largely focused on modifying the feature aggregation process and leveraging higher-order neighborhood information. Several approaches have been proposed to tackle the heterophily problem, where connected nodes may have dissimilar features or labels. H2GCN [21] introduced ego- and neighbor-embedding separation, along with the exploration of higher-order neighborhood structures. GPR-GNN [5] minimizes over-smoothing by integrating the PageRank method with GNNs. Furthermore, GGCN [20] addresses heterophily and over-smoothing issues by utilizing degree corrections and signed messages.

Interestingly, studies have revealed that basic models like Multi-Layer Perceptrons (MLPs) and LINK [10] can occasionally surpass conventional GNN architectures when dealing with heterophilic datasets. This observation has led to the development of hybrid methods that merge node features with graph-based representations. A prominent example is LINKX [10], which integrates MLPs for node features with LINK regression, showing promising performance on heterophilic graphs.

Another line of research has focused on aggregating features from neighbors at different distances. MixHop [1] and FSGNN [12] utilize the transition matrix's powers to capture multi-hop neighborhood information. FSGNN uses a regularizer method, such as softmax and L2-Normalization in GNN's layers.

Recent work has also explored novel ways to address the over-smoothing problem in deeper GNN architectures. Ordered GNN [16] proposes an approach that aligns the hierarchy of a rooted-tree with ordered neurons in node embeddings, effectively preserving information from different neighborhood depths.

While these advancements have significantly improved GNN performance on various graph types, there remains room for innovation in leveraging graph structure more effectively, particularly in the context of community detection and inter-community information flow. Our proposed method builds on these insights by incorporating spectral clustering and community hops, offering a novel approach to enhance GNN performance across diverse node-level predictions.

3 Preliminaries

In this section, we define the mathematical notations used in this study. Let $G = (V, E)$ represent an undirected input graph, where V denotes the set of nodes and $E \subseteq V \times V$ represents the set of edges. We use the adjacency matrix $A \in \{0, 1\}^{n \times n}$ to capture the graph's topological structure, where $A_{ij} = 1$ if $(i, j) \in E$ and $A_{ij} = 0$ otherwise.

To account for self-loops, we modify the adjacency matrix to $\tilde{A} = A + I$, where I denotes the identity matrix. The features of each node are now represented by a matrix $F \in \mathbb{R}^{n \times k}$, where k indicates the dimension of the feature space.

Additionally, we utilize the diagonal degree matrix D for the graph G , where $D_{ii} = d_i$ denotes the degree of node i , calculated by $d_i = \sum_j A_{ij}$. The normalized transition matrix P is then defined as $P = D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$.

3.1 Spectral Clustering

Spectral clustering is a robust method that utilizes the spectral properties of graph Laplacians to achieve clustering [11]. This section covers the essential matrices and concepts that are fundamental to spectral clustering techniques.

A key component in spectral clustering is the graph Laplacian, which comes in two primary forms. The unnormalized graph Laplacian is given by $L = D - W$, where W represents the weighted adjacency matrix and D is the diagonal degree matrix, with $D_{ii} = \sum_j w_{ij}$.

4 Begga et al.

Furthermore, the normalized graph Laplacian can be represented by the following formula: $\mathcal{L} = D^{-1/2}LD^{-1/2} = I - D^{-1/2}WD^{-1/2}$. Here, $\mathcal{L}$ provides a normalized version of the Laplacian that adjusts for the degree of nodes, facilitating more effective clustering.

These Laplacians have several important properties [6]:

1. They are symmetric and positive semi-definite.
2. The smallest eigenvalue is 0, with corresponding eigenvector $\mathbf{1}$ for L and $D^{1/2}\mathbf{1}$ for $\mathcal{L}$.
3. They have n non-negative, real-valued eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$.

For any vector $f \in \mathbb{R}^n$, we have:

$$f^T L f = \frac{1}{2} \sum_{i,j=1}^n w_{ij} (f_i - f_j)^2; \quad f^T \mathcal{L} f = \frac{1}{2} \sum_{i,j=1}^n w_{ij} \left(\frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}} \right)^2 \quad (1)$$

The multiplicity k of the eigenvalue 0 equals the number of connected components in the graph. The eigenspace of 0 is spanned by the indicator vectors of these components for L , and by $D^{1/2}$ -scaled indicator vectors for $\mathcal{L}$ [11].

Spectral clustering is closely related to the Normalized Cut (NCut) problem [15]. Given a partition of V into k disjoint subsets $A_1, \dots, A_k$, the NCut is defined as:

$$\text{NCut}(A_1, \dots, A_k) = \sum_{i=1}^k \frac{\text{cut}(A_i, V \setminus A_i)}{\text{vol}(A_i)} \quad (2)$$

where $\text{cut}(A, B) = \sum_{i \in A, j \in B} w_{ij}$ and $\text{vol}(A) = \sum_{i \in A} d_i$.

Minimizing NCut is NP-hard, but it can be relaxed to a tractable eigenvalue problem. This relaxation leads to the spectral clustering algorithm, which computes the first k eigenvectors $u_1, \dots, u_k$ corresponding to the k smallest eigenvalues of $\mathcal{L}$ (or generalized eigenvectors of $Lu = \lambda Du$) [11,15].

These eigenvectors form a matrix $U \in \mathbb{R}^{n \times k}$, where each row represents a node's k -dimensional embedding. This embedding enhances cluster properties in the data [11], allowing for easier separation in the new representation.

The final step involves clustering these embeddings, typically using the k -means algorithm, to obtain the approximate solution to the NCut problem. This approach effectively captures important graph properties such as communities and structural characteristics through the spectrum of the Laplacian, providing a powerful tool for graph partitioning [6,11,15].

3.2 Graph Neural Networks

Graph Neural Networks (GNNs) have emerged as a significant technique for handling data that is structured as graphs. These models adapt the concepts of convolutional neural networks to the non-Euclidean nature of graph data. The

core idea behind GNNs is to iteratively update node representations by collecting and processing information from neighboring nodes. A typical GNN layer can be formulated as:

$$H^{(i+1)} = \sigma(AH^{(i)}W^{(i)}), \quad (3)$$

where $H^{(i+1)}$ denotes the updated node features matrix at layer $i + 1$ after applying the layer transformation, $H^{(i)}$ represents the node features matrix before the transformation, $W^{(i)}$ is a matrix of learnable parameters and σ is a non-linear activation function. In the literature [9], it is common to use the normalized adjacency matrix, which is denoted as $\bar{A} = D^{-\frac{1}{2}}\hat{A}D^{-\frac{1}{2}}$.

4 Methodology

Our methodology addresses the limitations of existing GNN approaches by combining spectral graph theory with flexible multi-hop neighborhood aggregation. Figure 1 illustrates the difference between traditional hops and our community-based approach, which mitigates oversmoothing by emphasizing communal connections [4].

The foundation of our approach leverages spectral graph clustering to uncover global community structure. We begin with the eigendecomposition of the normalized Laplacian $\mathcal{L}$ [6]:

$$\mathcal{L} = U\Lambda U^T, \quad (4)$$

where U is the matrix of eigenvectors and Λ is the diagonal matrix of eigenvalues. The spectral properties of $\mathcal{L}$ are intimately connected to the graph's structure, with eigenvalues in the interval $[0, 2]$ [6].

We focus on the spectral gap, defined as $\gamma = \lambda_2 - \lambda_1$, where $\lambda_1 = 0$ and λ_2 is the smallest non-zero eigenvalue. This gap is related to the graph's connectivity and mixing time [17]. Specifically, the Cheeger constant $h(G)$, which measures the "bottleneckedness" of the graph, is bounded by the spectral gap through the Cheeger inequality:

$$\frac{\lambda_2}{2} \leq h(G) \leq \sqrt{2\lambda_2}, \quad (5)$$

This relationship, known as the Lovász bound [11,2], provides crucial insights into the graph's community structure. A small Cheeger constant indicates the presence of well-defined communities, while a large constant suggests a more uniformly connected graph [7].

We select the k leading eigenvectors corresponding to the smallest non-zero eigenvalues, where k is a hyperparameter. The choice of k can be guided by examining subsequent spectral gaps $(\lambda_{i+1} - \lambda_i)$, with a large gap suggesting a natural number of clusters [11].

To identify communities, we apply k -means clustering to the rows of the truncated eigenvector matrix U_k . This spectral embedding tends to separate nodes into more linearly distinguishable clusters than in the original graph space.

6 Begga et al.

We then introduce an edge-pruning mechanism to emphasize intra-community connections:

$$\mathcal{A}_{ij} = A_{ij} \cdot [C(i) = C(j)], \quad (6)$$

where $C(i)$ denotes the cluster assignment of node i . This pruning creates a block-diagonal structure in $\mathcal{A}$, aligning with the theoretical expectation of an ideal community structure in the spectral clustering framework. Following MixHop [1] and FSGNN [12], we will use the transition matrix to perform hops but this time with $\mathcal{A}$. Now the transition matrix can be defined as $\mathcal{P} = D^{-\frac{1}{2}} \mathcal{A} D^{-\frac{1}{2}}$

Building on this community-aware structure, we incorporate a multi-hop aggregation scheme with attention-like learnable parameters, inspired by recent GNN advancements [1,12]. The feature update rule for the $(i+1)$ -th layer is:

$$H^{(i+1)} = \left[\alpha_1 A H^{(i)} W_1^{(i)} \parallel \alpha_2 \mathcal{P}^1 H^{(i)} W_2^{(i)} \parallel \alpha_3 \mathcal{P}^2 H^{(i)} W_3^{(i)} \parallel \dots \parallel \alpha_{j+1} \mathcal{P}^j H^{(i)} W_{j+1}^{(i)} \right], \quad (7)$$

where j is the number of hops, $H^{(i)} \in \mathbb{R}^{n \times d_i}$ is the node feature matrix at the i -th layer, with n nodes and d_i features. $\mathcal{P} \in \mathbb{R}^{n \times n}$ is our community-aware transition matrix, and $W_{j+1}^{(i)} \in \mathbb{R}^{d_i \times d_{out}}$ are learnable weight matrices for each hop distance j at layer i , and $\parallel$ denotes column-wise concatenation.

We introduce learnable attention-like parameters α_{j+1} for each hop embedding and the original node features and adjacency, allowing the model to weigh the importance of different neighborhood scales adaptively. Importantly, these attention parameters are constrained to sum to 1: $\sum_{k=1}^{j+1} \alpha_k = 1, \quad \alpha_k \geq 0 \quad \forall k \in \{0, 1, \dots, m\}$:

This constraint ensures that the attention mechanism is a proper weighting system across different hop distances.

This multi-hop aggregation allows the model to simultaneously capture and weigh information from various neighborhood scales, as we illustrate in Figure 2. For instance, if $j = 2$, the model considers the initial adjacency ($\alpha_1 A H^{(i)} W_1^{(i)}$), its immediate neighbors ($\alpha_2 \mathcal{P}^1 H^{(i)} W_2^{(i)}$), and its 2-hop neighbors ($\alpha_3 \mathcal{P}^2 H^{(i)} W_3^{(i)}$) in each layer. The use of different weight matrices $W_{j+1}^{(i)}$ and attention parameters α_{j+1} for each hop distance and the initial adjacency, enables the model to learn the relative importance of information from different scales while maintaining a balanced aggregation.

This approach generalizes the power iteration method often used in spectral clustering, allowing the capture of higher-order relationships in the graph while maintaining the ability to differentiate between local and global structural information. By learning to assign different importance to various neighborhood scales and preserving the original feature information, our model effectively captures complex patterns of node similarity and dissimilarity, adapting to both homophilic and heterophilic graph structures.

Community-Hop: Enhancing Node Classification 7

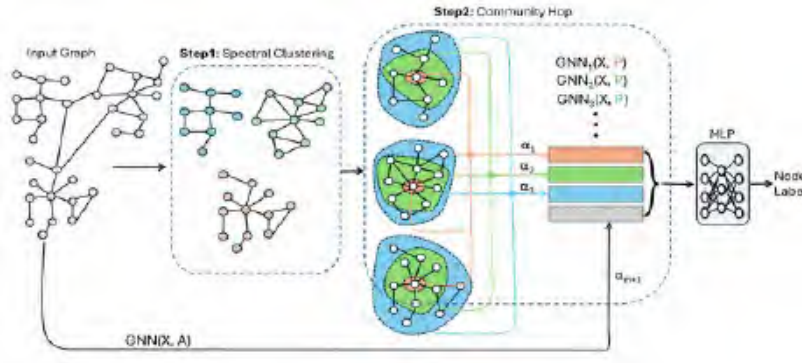


Fig. 2: Illustration of the spectral clustering and GNN propagation process. The input graph undergoes spectral clustering to identify communities (Step 1). Then, a community-aware multi-hop aggregation is performed (Step 2), where information is propagated within communities. The obtained node representations are concatenated and then passed through a MLP for the final prediction of node labels.

4.1 Computational Complexity

Our approach's computational complexity is divided into two primary processes: preprocessing and processing.

The preprocessing phase involves calculating the k leading eigenvectors of the normalized Laplacian matrix. For a graph with n nodes and m edges, this computation has a worst-case time complexity of $O(n^3)$ and a space complexity of $O(n^2)$. However, performance can be enhanced by employing optimized algorithms tailored for sparse graphs.

During the processing phase, k -means clustering and GNN propagation are performed. This step has a time complexity is $O(nk^2 + LHmF)$, where k denotes the number of clusters and the dimension of spectral embedding, L is the number of GNN layers, H represents the number of hops, and F is the number of features. The space complexity for the processing phase is $O(n(k + F) + m)$.

Despite the preprocessing step being computationally intensive, especially for larger graphs, it provides a comprehensive basis for identifying global community structures. This trade-off between computational cost and structural insight allows our approach to effectively capture both global and local patterns in the graph, enabling robust performance on both homophilic and heterophilic graph structures.

5 Experiments and discussions

This section evaluates the proposed method's performance on six node classification benchmarks. The experimental results reveal that the Community-HOP

8 Begga et al.

	TEXAS	WISCONSIN	CORNELL	CITeseer	PUBMED	CORA
HOM LEVEL	0.11	0.21	0.30	0.74	0.80	0.81
# NODES	183	251	183	3,327	19,717	2,708
# EDGES	295	466	280	4,676	44,324	5,278
# CLASSES	5	5	5	7	3	6
MLP	80.81 $\pm$ 4.75	85.29 $\pm$ 6.40	81.89 $\pm$ 6.40	74.02 $\pm$ 1.90	75.69 $\pm$ 2.00	87.16 $\pm$ 0.37
GCN [9]	55.14 $\pm$ 5.16	51.76 $\pm$ 3.06	60.54 $\pm$ 5.30	76.50 $\pm$ 1.36	88.42 $\pm$ 0.50	86.98 $\pm$ 1.27
GAT [18]	52.16 $\pm$ 6.63	49.41 $\pm$ 4.09	61.89 $\pm$ 5.05	76.55 $\pm$ 1.23	87.30 $\pm$ 1.10	86.33 $\pm$ 0.48
GRAPHSAGE [8]	82.43 $\pm$ 6.14	81.18 $\pm$ 5.56	75.95 $\pm$ 5.01	76.04 $\pm$ 1.30	88.45 $\pm$ 0.50	86.90 $\pm$ 1.04
H2GCN [21]	84.86 $\pm$ 7.23	87.65 $\pm$ 4.89	82.70 $\pm$ 5.28	77.11 $\pm$ 1.57	89.49 $\pm$ 0.38	87.87 $\pm$ 1.20
GEOM-GCN [13]	66.76 $\pm$ 2.72	64.51 $\pm$ 3.66	60.54 $\pm$ 3.67	78.02 $\pm$ 1.15	89.95 $\pm$ 0.47	85.35 $\pm$ 1.57
LINKX [10]	74.60 $\pm$ 8.37	75.49 $\pm$ 5.72	77.84 $\pm$ 5.81	73.19 $\pm$ 0.99	87.86 $\pm$ 0.77	84.64 $\pm$ 1.13
GGCN [20]	84.86 $\pm$ 4.55	86.86 $\pm$ 3.29	85.68 $\pm$ 6.63	77.14 $\pm$ 1.45	89.15 $\pm$ 0.37	87.95 $\pm$ 1.05
CGNN [19]	71.35 $\pm$ 4.05	74.31 $\pm$ 7.26	66.22 $\pm$ 7.69	76.91 $\pm$ 1.81	87.70 $\pm$ 0.49	87.10 $\pm$ 1.35
MixHop [1]	77.84 $\pm$ 7.73	75.88 $\pm$ 4.90	73.51 $\pm$ 6.34	76.26 $\pm$ 1.33	85.31 $\pm$ 0.61	87.61 $\pm$ 0.85
FSGNN [12]	87.30 $\pm$ 5.29	87.84 $\pm$ 3.37	85.13 $\pm$ 6.07	77.40 $\pm$ 1.90	77.40 $\pm$ 1.93	87.93 $\pm$ 1.00
GPRGNN [5]	78.38 $\pm$ 4.36	82.94 $\pm$ 4.21	80.27 $\pm$ 8.11	77.13 $\pm$ 1.67	87.54 $\pm$ 0.38	87.95 $\pm$ 1.18
COMMUNITY-HOP	89.46 $\pm$ 5.72	89.01 $\pm$ 3.84	82.70 $\pm$ 3.00	78.30 $\pm$ 2.13	89.50 $\pm$ 0.47	88.22 $\pm$ 1.29

Table 1: Node-classification accuracies. Top three models are highlighted: **First**, **Second**, **Third**.

method achieved the highest performance on four out of six datasets compared to baselines. This section describes the datasets and experimental settings, followed by a comprehensive comparison of the results and a detailed analysis.

To evaluate the efficacy of the Community-HOP, we selected six small to medium real-world node classification benchmark datasets: Cora, Cornell, PubMed, Texas, and Wisconsin [14]. A statistical summary of these datasets, including the edge homophily ratio (HOM LEVEL) [21], offers insight into the dataset’s heterophily. A higher HOM LEVEL indicates greater heterophily, posing a challenge for vanilla GNN models, which typically perform worse under these conditions.

For the node classification experiments, we utilized the dataset splits provided by [14]. Each split includes 48% of the data for training, 32% for validation, and 20% for testing. The performance metrics are reported as the average accuracy with standard deviation across 10 different splits. All models were trained for a total of 3000 epochs using the Adam optimizer and cross-entropy loss function. To optimize the Community-HOP method, hyperparameter tuning was carried out, focusing on parameters such as learning rate, dropout rate, number of clusters, number of hops, and hidden dimensions. A grid search strategy was used to examine different combinations of these hyperparameters. Detailed information on the hyperparameter settings for each dataset is available at the following link.

Our experimental findings show that Community-HOP markedly exceeds the performance of existing methods in accuracy across four diverse datasets, showcasing its effectiveness and adaptability in node classification tasks with varying degrees of heterophily. The analysis of edge homophily ratios (HOM LEVEL) emphasizes the difficulties encountered with higher heterophily levels

and highlights the improvements offered by Community-HOP over conventional GNNs and other state-of-the-art techniques.

However, our method does not achieve superior performance on the Cornell and PubMed datasets. In the case of Cornell, this limitation can be attributed to difficulties in accurately computing spectral clusters, exacerbated by significant gaps in the dataset's homophily structure. For PubMed, the high volume of nodes and edges impedes our ability to effectively capture the underlying community structure. Consequently, inadequate clustering results in suboptimal performance for community-specific hops.

Our method focuses on community nodes by executing multiple hops within the community and shows promising results, particularly in homophilic environments where neighbors share the same label. This characteristic is advantageous as it aligns with the assumption that nodes within such environments exhibit high intra-community homophily. Notably, our Community-HOP also demonstrates effective performance in heterophilic contexts, suggesting that it can adeptly manage heterophily within communities. This adaptability contributes to its overall improved classification performance across various datasets, showing its potential for broader applicability in diverse graph-based tasks.

6 Conclusion and Future Work

In this study, we introduced a Community-HOP-based GNN model designed to address the challenge of heterophily in graphs. Central to our approach is the Community-Hop method, which leverages community structural information to refine the feature aggregation process within the GNN layers. This technique enhances the relevance of information flow both within and across communities by integrating spectral community detection with an adapted transition matrix for inter-community hops. However, a significant limitation of our approach is its computational cost, particularly for large-scale graphs and the determination of an optimal parameter k . Future work will focus on addressing these challenges by advancing spectral clustering techniques and developing methods for automatic optimization of k .

Acknowledgement

The authors acknowledge the support from the Spanish Government under project PID2022-142516OB-I00.

References

1. Abu-El-Haija, S., Perozzi, B., Kapoor, A., Alipourfard, N., Lerman, K., Harutyunyan, H., Ver Steeg, G., Galstyan, A.: Mixhop: Higher-order graph convolutional architectures via sparsified neighborhood mixing. In: international conference on machine learning (2019)

10 Begga et al.

2. Arnaiz-Rodríguez, A., Begga, A., Escolano, F., Oliver, N.: Diffwire: Inductive graph rewiring via the lovász bound. In: Learning on Graphs Conference, LoG 2022, 9-12 December 2022, Virtual Event. Proceedings of Machine Learning Research (2022)
3. Bianchi, F.M., Grattarola, D., Alippi, C.: Mincut pooling in graph neural networks. CoRR **abs/1907.00481** (2019)
4. Cai, C., Wang, Y.: A note on over-smoothing for graph neural networks. CoRR **abs/2006.13318** (2020)
5. Chien, E., Peng, J., Li, P., Milenkovic, O.: Adaptive universal generalized pagerank graph neural network. In: International Conference on Learning Representations (2021)
6. Chung, F.R.: Spectral Graph Theory. American Mathematical Society (1997)
7. Deng, S., Ling, S., Strohmer, T.: Strong consistency, graph laplacians, and the stochastic block model. J. Mach. Learn. Res. **22**, 117:1–117:44 (2021)
8. Hamilton, W.L., Ying, R., Leskovec, J.: Inductive representation learning on large graphs. In: Proceedings of the 31st International Conference on Neural Information Processing Systems (2017)
9. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: International Conference on Learning Representations (ICLR) (2017)
10. Lim, D., Hohne, F.M., Li, X., Huang, S.L., Gupta, V., Bhalerao, O.P., Lim, S.N.: Large scale learning on non-homophilous graphs: New benchmarks and strong simple methods. In: Advances in Neural Information Processing Systems (2021)
11. von Luxburg, U.: A tutorial on spectral clustering. CoRR **abs/0711.0189** (2007)
12. Maurya, S.K., Liu, X., Murata, T.: Improving graph neural networks with simple architecture design (2021)
13. Pei, H., Wei, B., Chang, K.C., Lei, Y., Yang, B.: Geom-gcn: Geometric graph convolutional networks. CoRR **abs/2002.05287** (2020)
14. Pei, H., Wei, B., Chang, K.C.C., Lei, Y., Yang, B.: Geom-gcn: Geometric graph convolutional networks (2020)
15. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **22**(8), 888–905 (2000)
16. Song, Y., Zhou, C., Wang, X., Lin, Z.: Ordered GNN: Ordering message passing to deal with heterophily and over-smoothing. In: The Eleventh International Conference on Learning Representations (2023)
17. Topping, J., Giovanni, F.D., Chamberlain, B.P., Dong, X., Bronstein, M.M.: Understanding over-squashing and bottlenecks on graphs via curvature. In: International Conference on Learning Representations (2022)
18. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph Attention Networks. International Conference on Learning Representations (2018)
19. Yamamoto, T.: Crystal graph neural networks for data mining in materials science (2019)
20. Yan, Y., Hashemi, M., Swersky, K., Yang, Y., Koutra, D.: Two sides of the same coin: Heterophily and oversmoothing in graph convolutional neural networks. CoRR **abs/2102.06462** (2021)
21. Zhu, J., Yan, Y., Zhao, L., Heimann, M., Akoglu, L., Koutra, D.: Beyond homophily in graph neural networks: Current limitations and effective designs. In: Proceedings of the 34th International Conference on Neural Information Processing Systems (2020)

Originally published in *Pattern Recognition Letters* 186, pp. 71-77, Elsevier, September 2024, DOI 10.1016/j.patrec.2024.09.009.

1



Pattern Recognition Letters
journal homepage: www.elsevier.com

Hierarchical Glocal Attention Pooling for Graph Classification

Waqar Ali^{a,b,**}, Sebastiano Vascon^{a,c}, Thilo Stadelmann^{b,c}, Marcello Pelillo^a

^aCa' Foscari University of Venice, Via Torino, 155, Mestre, Venice, 30170, Italy

^bZHAW Centre for Artificial Intelligence, Technikumstrasse 9, Winterthur, 8401, Switzerland

^cECLT European Centre for Living Technology, Sestiere Dorsoduro, 3911, Venice, 30123, Italy

ABSTRACT

Graph pooling is an essential operation in Graph Neural Networks that reduces the size of an input graph while preserving its core structural properties. Existing pooling methods find a compressed representation considering the Global Topological Structures (e.g., cliques, stars, clusters) or Local information at node level (e.g., top- k informative nodes). However, an effective graph pooling method does not hierarchically integrate both Global and Local graph properties. To this end, we propose a dual-fold Hierarchical Global Local Attention Pooling (HGLA-Pool) layer that exploits the aforementioned graph properties, generating more robust graph representations. Exhaustive experiments on nine publicly available graph classification benchmarks under standard metrics show that HGLA-Pool significantly outperforms eleven state-of-the-art models on seven datasets while being on par for the remaining two.

© 2024 Elsevier Ltd. All rights reserved.

1. Introduction

Graph Neural Networks (GNNs) have recently gained significant attention due to their ability to process graph-structured data effectively. They have shown effectiveness in various tasks of classifying graphs and learning graph representations (Kipf & Welling, 2016), including understanding and predicting interactions between molecules and proteins (Wang et al., 2022; Réau et al., 2023a) that can lead to significant advancements in drug discovery, analyzing structure and dynamics of interactions in social networks (Khemani et al., 2024), operating on knowledge graphs to enable Retrieval-Augmented Generation (RAG) (Mavromatis & Karypis, 2024) or enable the detection of visual objects from their context (Han et al., 2022), e.g., to support document accessibility (Schmitt-Koopmann et al., 2022). Within the scope of graph classification, the Graph Pooling (GP) methods play a pivotal role in GNN architectures that map the set of nodes or subgraphs into a compact representation to capture a meaningful structure of the entire graph (Pang et al., 2021). Early GP methods, referred to as global methods (Bai

et al., 2019; Simonovsky & Komodakis, 2017), are the simplest approaches, which reduce the size of the input graph by performing a sum or average of all nodes without considering the hierarchical information of the graph and may lose feature information. Recently, several advanced hierarchical GP methods, such as node cluster and node selection methods (Chen & Gel, 2023; Gao & Ji, 2019; Luzhnica et al., 2019), have been proposed to tackle the limitations of global pooling and obtain state-of-the-art performance.

The node cluster methods like DiffPool (Ying et al., 2018) and CliquePool (Luzhnica et al., 2019) capture the Global Topological Structure (GTS) of a graph, which extracts the overall architecture and connectivity patterns of the entire network by partitioning the graph into clusters. Then, each cluster transfers into a supernode to create a coarsened graph. However, the CliquePool (Luzhnica et al., 2019) method has limitations in extracting overlapping nodes between cliques of the same size and treats all cliques equally informative without considering node features. On the other hand, node selection methods such as TopKPool, SAGPool, and Topological Pooling (Xu et al., 2022; Lee et al., 2019; Chen & Gel, 2023) identify the most significant nodes based on their feature values or attention scores while dropping unnecessary nodes. These methods primarily focus on the graph's Local Topological Structure (LTS), which includes the analysis of individual nodes, their attributes,

^{**}Corresponding author:

e-mail: waqar.ali@unive.it (Waqar Ali),
sebastiano.vascon@unive.it (Sebastiano Vascon), stdm@zhaw.ch
(Thilo Stadelmann), pelillo@unive.it (Marcello Pelillo)

and neighborhoods. However, there is a lack of an effective GP method that combines the graph's global and local properties hierarchically to improve the graph representation.

To address the limitations of existing GP methods, this study proposes Hierarchical GlobalLocal Attention Pooling (HGLA-Pool), a novel pooling method designed to capture graphs' local and global properties. The proposed approach is structured as a two-fold process, each fold addressing specific limitations of existing GP approaches. The fold-1 leverages the idea of cliques that perform clique pooling (Luzhnica et al., 2019), which aims to encapsulate the graph's GTS. In this fold, we design a rule-based method to enhance the clique representation by identifying the overlapping nodes between cliques. Identifying overlapping nodes is crucial in various real-world scenarios, such as molecular biology, where a molecule might serve as a shared binding site for multiple proteins, and in social network analysis, where individuals often belong to multiple social circles, including professional networks and personal connections (Wang et al., 2022; Khemani et al., 2024). Additionally, we introduce a novel dynamic fusion method that incorporates graph structure information and node features to calculate scores for each clique and select the topk significant cliques using this fusion score. Furthermore, we recognize that every node within a selected clique does not contribute equally to the graph representation during the pooling operation. We argue that the most informative nodes from all ranked cliques should be captured in GP. Therefore, in fold-2, we develop a LocalPool layer on top of fold-1 using multi-attention to pinpoint and emphasize the most informative nodes within the ranked cliques, ensuring a more meaningful representation of the overall graph for pooling operation. The hierarchical architecture of our method, as illustrated in Figure 1, reflects this dual-fold approach. We summarized our contributions as follows:

- This study introduces HGLA-Pool, a novel pooling layer that sequentially integrates the global structural properties of the graph with the local node's properties.
- We develop a rule-based method to extract the overlapping nodes between cliques. We also design a novel dynamic fusion method that leverages graph structural information and node features to identify the most relevant global structures (cliques). Furthermore, this study develops a multi-attention LocalPool to capture the local node's properties.
- We experimentally prove that considering both sources of information (global and local) yields a better-learned representation. We consistently outperformed 11 state-of-the-art models on seven diverse and challenging benchmarks.

2. Related Work

Related work includes existing GP studies focusing on global and hierarchical techniques. Global pooling methods usually use sum or mean functions to aggregate the node features that generate a single vector representation for the entire graph. For example, Vinyal et al. (Vinyals et al., 2015) proposed a Set2Set framework that uses the Long Short-Term Memory model to generate graph representation by identifying informative nodes.

Authors (Zhang et al., 2018) developed a novel GNN layer to capture multi-level node features and a sort pooling that sorts these features to keep more graph information. However, global methods ignore the graph's hierarchical information during the pooling operations. Meanwhile, hierarchical methods leverage graph structures and node features for the pooling. Based on designing properties, the hierarchical methods can be classified into two main classes: node selection and node cluster.

Node selection methods reduce the graph size by selecting the most informative nodes based on their node features or attention scores. Hongyang et al. (Gao & Ji, 2019) developed a TopkPool that uses node scalar values on a trainable projection vector as the node score. SAGPool (Lee et al., 2019) further adopts the GNN layer to calculate node scores. Jinheon et al. (Xu et al., 2022) proposed Multistructure Attention Convolutional (MAC) pooling that incorporates dual-node scoring strategies to obtain the importance of nodes. These methods are considered more computationally efficient as they only require calculating an informative node's score, but they ignore the graph's GTS information. The node cluster method captures the global properties by finding the clusters within the graph. DiffPool (Ying et al., 2018) uses the GNN layer output to learn a differentiable soft cluster assignment to extract the clusters. In (Luzhnica et al., 2019), the authors introduced a CliquePool method to learn the GTS by extracting the maximal cliques from the graph. Quasi-CliquePool (Ali et al., 2023) further improves the CliquePool by capturing the overlapping nodes between two cliques using the replicator dynamic algorithm. (Islam et al., 2023) developed a Motif-based pooling method that extracts the high-order graph structure by combining cluster and selection pooling operations. The motif method treated all motif structures equally without considering the node features. Graph Multiset Transformer (GMT) pooling incorporates node structural dependencies with a multi-head attention mechanism to enhance the graph representation by identifying node interaction. Jinlong et al. (Du et al., 2021) design a Multi-channel pooling (MuchPool) that combines TopkPool and DiffPool methods to create a more comprehensive hierarchical representation of graphs. However, the MuchPool method suffers from considerable computational complexity due to the simultaneous operation of three distinct channels to capture node feature information, global and local structural information. A recent study (Chen & Gel, 2023) introduced a Wit-TopoPool, which integrates a witness complex-based topological embedding mechanism with a global pooling layer. This approach aims to extract comprehensive and discriminative topological information from graphs.

In conclusion, the current landscape in pooling methods demonstrates a notable gap: the lack of an effective pooling approach that sequentially combines node selection and node cluster techniques to yield a robust graph representation. Addressing this gap, we proposed a novel HGLA-Pool method that captures the graph's local and global properties with node features to generate a more robust graph representation.

3. Methodology

This section explains our proposed HGLA-Pool, shown in Figure. 1. The HGLA-Pool performs pooling operations in two-fold, capturing the graph's global and local properties and then aggregating the results to form a new pooled graph. The following sections present a detailed description of the proposed HGLA-Pool.

3.1. Preliminaries

This section defines mathematical notations and concepts about GNNs, which will be used in this study.

We represent the input graph as $G = (V, E)$, where V denotes the nodes and E the edges in the graph. The connection between nodes of G can be represented by an adjacency matrix $A \in \mathbb{R}^{N \times N}$, where $N = |V|$ is the number of nodes. The matrix $X \in \mathbb{R}^{N \times d}$ represents the node features matrix, where d is the dimension of the feature space. The GP operation changes the number of nodes at each pooling layer, thus, we further identify the graph at the l -th layer as $G^{(l)} = (V^{(l)}, E^{(l)})$. The node feature and adjacency matrices of the pooled graph represent as $X^{(l)} \in \mathbb{R}^{N^{(l)} \times d^{(l)}}$ and $A^{(l)} \in \mathbb{R}^{N^{(l)} \times N^{(l)}}$.

Given a graph dataset $D = \{(G_1, Y_1), (G_2, Y_2), \dots\}$, where G_i and Y_i are the i th graphs and the set of labels corresponding to graphs, respectively. The principle of the graph classification task is to learn a mapping function, such as $f: G \rightarrow Y$.

Several GNN architectures have been developed, such as Graph Convolutional Network (GCN) Kipf & Welling (2016) and Graph Attention Network (GAT) Veličković et al. (2017). They can learn node representations by propagating and aggregating information from a node and its neighbors, achieving outstanding performance in various graph-related tasks. Therefore, this study adopts hybrid graph convolution operators to extract significant cliques (i.e., a graph is a subset of nodes such that every two distinct nodes are adjacent) and node representations from multiple perspectives. In the following, we briefly outline the mechanisms of GCN (Kipf & Welling, 2016) and GAT (Veličković et al., 2017).

GCNs learn node representations by considering topology information and node features in a graph. The basic operation of a GCN layer can be described as follows:

$$X^{(l+1)} = \sigma(\hat{D}^{-1/2} \hat{A} \hat{D}^{-1/2} X^{(l)} W^{(l)}) \quad (1)$$

where $\hat{A}$ is the adjacency matrix with added self-loops, $\hat{D}$ is the degree matrix, $X^{(l)}$ is the feature matrix at layer l , $W^{(l)}$ is the trainable weight matrix, and σ is an activation function.

While GATs introduce an attention mechanism to GNNs, allowing the model to assign different importance to different nodes within a neighborhood. The expression of GAT is given as follows:

$$X^{(l+1)} = \sigma \left(\frac{1}{K} \sum_{k=1}^K \sum_{j \in N(i)} a_{ij}^k W^k X_j \right) \quad (2)$$

where K represents the number of heads, X_j is the feature vector of node v_j , which is a neighbor of node v_i , a_{ij}^k is the attention weight between X_i and X_j , and W^k is a trainable weight matrix for head k .

3.2. Fold 1: Global Topological Structure Learning

In fold 1, we aim to capture the GTS information of the graph. For this purpose, we use the CliquePool (Luzhnica et al., 2019) with a modified version of the Bron-Kerbosch algorithm (Cazals & Karande, 2008) to extract all cliques from the input graph G , denoted as a set $C = \{c_1, c_2, \dots, c_t\}$. This fold is designed to overcome two significant challenges of CliquePool (Luzhnica et al., 2019): extracting overlapping nodes between cliques of the same size and ranking cliques based on their node features.

Overlapping Nodes Extraction: Overlapping nodes substantially impact various real-world scenarios, including molecular biology, where a particular molecule may act as a shared binding site for multiple proteins, and in social network analysis, a person may be part of both a professional network (Wang et al., 2022; Réau et al., 2023b). We design a rule-based algorithm using three conditions to handle nodes that belong to multiple cliques during the clique extraction process: 1) If a node is already assigned to a clique, it is only added to the current clique being considered if the sizes of the existing and current clique are equal or current clique size is bigger than existing clique in terms of nodes, 2) If a node is connected to multiple cliques, in that case, we evaluate whether this specific node has maximum connectivity with the nodes of an associated clique. If it does and is not already part of the associated clique, the node is integrated into that clique, and 3) If nodes of a clique have already been assigned to larger cliques, then we remove that clique. This criterion ensures that the node's association with the clique is not arbitrary but is based on a meaningful and substantial connectivity pattern. In Figure. 1, the fold-1 demonstrates the extraction of overlapping nodes; see the cliques 1, 2, 3, and 4 with overlapping nodes.

The existing clique pooling method (Luzhnica et al., 2019) partitions the graph into cliques based on the graph's topological structure without considering the node features. Furthermore, this method treats all cliques equally, which can be problematic because not all cliques are equally significant; some are rich in information, and others might merely consist of a single node, thus carrying significantly less information to subsequent pooling layers. To solve these issues, we developed a novel dynamic fusion method to calculate each clique's importance using graph structure information and node features.

Graph Structure Score: The graph structure can provide useful information on the importance of nodes, such as node degrees and the shortest paths between different nodes. Hence, we borrow the node centrality definition to illustrate each clique's importance within the graph. We introduce a heuristic method to calculate a score for each clique using graph structure information: fusion of degree centrality and unique neighbor count. In the first step, we compute the mean degree centrality $\bar{D}_{c_i}$ of all nodes in c_i . Next, we calculate the number of unique neighbors $\bar{U}_{c_i}$ of a clique c_i by collecting all neighbors of nodes in c_i and removing any nodes that are part of this clique itself. Finally, the structure score S_{c_i} of each clique is derived as a weighted sum of these two $\bar{D}_{c_i}$ and $\bar{U}_{c_i}$ metrics, parameterized by β and calculated as follows:

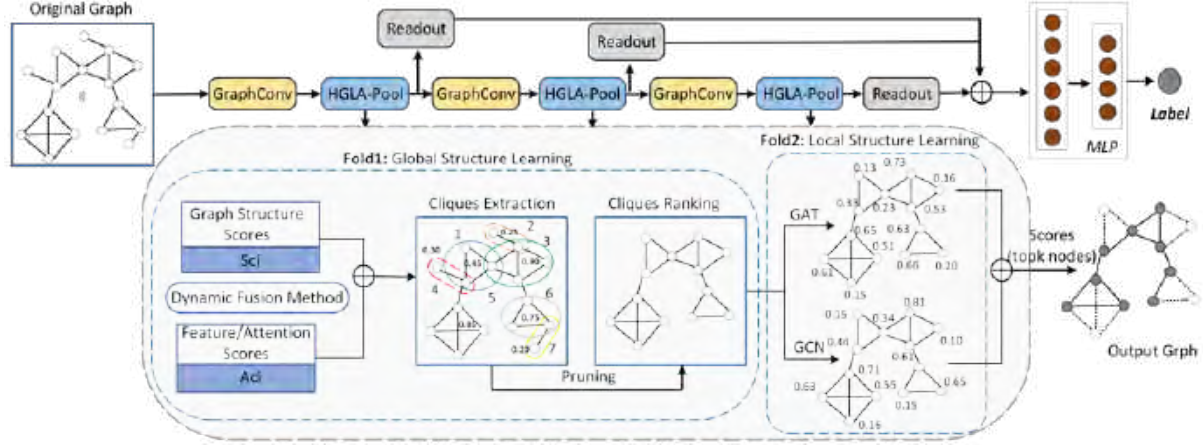


Fig. 1: The hierarchical architecture of the HGLA-Pool integrates with GNN for graph classification. The grey lightbox demonstrates the overall workflow of HGLA-Pool, which contains two folds. The **fold-1** performs three steps: 1) capture GTS by finding all maximal cliques within the graphs; 2) find overlapping nodes between two cliques to obtain more reasonable clique representations; and 3) incorporate the dynamic fusion method to rank the cliques and form a new pooled graph. Meanwhile, the **fold-2** takes this pooled graph as input and captures the LTS by selecting several significant nodes from ranked cliques based on scoring criteria (e.g., GCN and GAT). The readout function is applied after each pair of GNN and HGLA-Pool layers to generate the graph-level representation.

$$\bar{D}_{c_i} = \frac{1}{|c_i|} \sum_{v \in c_i} d_v; \quad \bar{U}_{c_i} = |\cup_{v \in c_i} N(v) - c_i| \quad S_{c_i} = \beta \times \bar{D}_{c_i} + (1 - \beta) \times \bar{U}_{c_i}; \quad (3)$$

Node Feature Score: In addition to topological information, the node features information contributes significantly to defining the graph properties (Kong et al., 2023). For instance, in chemical molecules, node features represent atom types, which are essential for predicting the graph's characteristics. Therefore, it can be an effective resource to show the clique's importance within a graph. For this purpose, we use a graph convolution layer to calculate importance scores for each node. The calculation procedure is outlined as follows:

$$Z(v) = \sigma(\bar{D}^{-1/2} \tilde{A} \bar{D}^{-1/2} X W); \quad Z_{c_i} = \sum_{v \in c_i} Z(v) \quad (4)$$

where $Z(v) \in \mathbb{R}^{N \times 1}$ represents the importance scores and $\sigma(\cdot)$ is a nonlinear activation function (e.g. softmax). The $\tilde{A} \in \mathbb{R}^{N \times N} = A + I$ shows the adjacency matrix with self-loops, $\bar{D} \in \mathbb{R}^{N \times N} = \sum_j \bar{A}_{ij}$ is the degree matrix of $\tilde{A}$ and $W \in \mathbb{R}^{d \times 1}$ is a trainable weight parameter. Various aggregation functions like max and sum can be employed to calculate the score for each clique. In our implementation, we choose the sum function to aggregate the importance scores of the nodes within the corresponding clique c_i . Given a clique c_i , the importance clique score Z_{c_i} is computed in equation 4.

Fusion Score: To achieve a more robust score for each clique, it is necessary to use both graph structure and node feature information fully. So, we combine S_{c_i} with the Z_{c_i} score using a learnable parameter α to obtain a fuse score F_{c_i} (see leftmost side of equation 5), thus providing a more comprehensive representation of the graph's topology. After obtaining the F_{c_i} scores, we used them to select cliques that the pooling operator should preserve. In detail, we first rank the cliques according

to their fusion score F_{c_i} , then a subset of top-ranked cliques are selected for constructing the pooled graph as follows:

$$F_{c_i} = \alpha \times Z_{c_i} + (1 - \alpha) \times S_{c_i}; \quad C'_j = \text{rank}(F_{c_i}, \lceil C_r \times |C| \rceil) \quad (5)$$

where $\text{rank}(\cdot)$ denotes the function that returns the important cliques, C_r is the clique ratio, and $|C|$ is the total cliques. We update the node feature based on selected clique nodes (see leftmost side of equation 6). We construct a new adjacency matrix A' where only the edges between the nodes in the selected cliques are retained. For this purpose, we define an edge mask matrix E_m , which is structured to correspond with the A using node mask M_i . Each element in E_m is set to 1 if both corresponding nodes (v_i and v_j in M_i) are part of the selected cliques and set to 0 otherwise. This can be represented mathematically as:

$$X' = X \odot M; \quad A' = A \odot E_m \quad (6)$$

where X' is the new node feature matrix, X is the original node feature matrix, $\odot$ represents the broadcasted element-wise product, and M is a vector where each entry corresponds to a node in the graph. The dimension of M is the same as the number of nodes in the graph. Each entry in M specifies whether or not the corresponding node is part of a selected clique (1 for nodes in C'_k and 0 for all other nodes). A' is the new adjacency matrix, and A is the original edge index matrix. The first fold's output serves as an input for the subsequent fold. The information provided by this fold acts as a contextual framework that enables the second fold to identify local structures effectively.

3.3. Fold 2: Local Topological Structure Learning

This section presents how our proposed LocalPool layer leverages multi-attention mechanisms to learn LTS information by extracting the most informative nodes from ranked cliques. Existing pooling methods like CliquePool and Kplex-Pool (Luzhnica et al., 2019; Bacciu et al., 2021) extract all possible cliques from the input graph and often transfer each clique

into a supernode to form a pooled graph. This transformation aims to reduce the graph size and complexity by aggregating the information within each clique. However, this transformation does result in a loss of information at the individual node level within each clique. The fine-grained details of interactions and relationships between nodes within a clique are not preserved in the pooled representation. This loss of information can potentially affect the downstream tasks, especially if the node-level information is crucial for classification tasks.

To address this, we propose a LocalPool layer to refine the graph further by selecting the most important nodes from ranked cliques. This refinement and selection process allows our pooling operation to help preserve the key information within each clique while reducing the graph size and complexity. Our LocalPool approach significantly diverges from existing methods like SAGPool (Lee et al., 2019). Specifically, SAGPool relies on a single strategy for computing node importance, yielding less robust node rankings. Inspired by the MAC method Xu et al. (2022), we adopt a dual-strategy-based pooling layer to calculate node importance, as illustrated in Figure 1. First, we employ a GCN layer to calculate importance scores for the nodes. This choice stems from the GCN model's ability to learn node representations by aggregating information from neighboring nodes, offering a localized perspective. GCNs aggregate feature information from a node's local neighborhood, providing a stable and robust representation by averaging the features of neighboring nodes. Simultaneously, we employ the Graph GAT layer to compute attention scores. GATs introduce an attention mechanism that assigns different importance to different nodes within a neighborhood, capturing more intricate relationships. GATs can adapt to varying graph structures by learning the attention coefficients, making them versatile and capable of capturing complex patterns and dependencies that might be missed by the uniform aggregation of GCNs. Given that GCN and GAT extract graph information from distinct viewpoints—GCN provides a robust, averaged view of a node's neighborhood, and GAT offers a nuanced, dynamically weighted view. we consider this process a multi-attention strategy for calculating node importance scores. By aggregating the scores from both GCN and GAT (using operations such as sum or max), we achieve a more comprehensive and robust node importance score, which enables our pooling method to select the most important nodes. We consider this process as a multi-attention strategy, and it is calculated as follows:

$$S_{N_i} = \text{Aggregation} \left(\sigma \left(\frac{1}{K} \sum_{k=1}^K \sum_{j \in N(i)} a_{ij}^k W^k X_j' \right), \sigma \left(\bar{D}^{-1/2} \bar{A}' \bar{D}^{-1/2} X' W \right) \right) \quad (7)$$

where S_{N_i} represents a multi-attention score for a node, and Aggregation is an operation such as max, mean, and sum. We use the max function to select the maximum value from each row (e.g., for each node, choose the maximum score from either the GCN or GAT model). The $W^k \in \mathbb{R}^{d \times 1}$ and $W \in \mathbb{R}^{d \times 1}$ are parameter weight matrices and used for linear transformations of node features. We identify and select high-score nodes from the output graph of Fold-1 to construct a pooled graph using the scores S_{N_i} . The details of this process are as follows:

$$idx = \text{topk}(S_{N_i}, [P_r * N']); \quad X^{l+1} = X'(idx, :) \odot S_{N_i}(idx, :); \quad A^{l+1} = A'(idx, idx) \quad (8)$$

where topk is the function that returns the indices of the top $N^{l+1} = [P_r * N']$ values. The $X'(idx, :)$ perform the row-wise (node-wise) indexed node feature matrix, $S(idx, :)$ represents the row-wise indexed node importance score matrix, and $A'(idx, idx)$ is the row-wise and col-wise indexed adjacency matrix. X^{l+1} and A^{l+1} represent the new node feature and adjacency matrices, respectively.

3.4. Hierarchical HGLA-Pool Architecture and Readout Layer

We employ a hierarchical pooling architecture, integrating multiple GCN layers with HGLA-Pool layers to perform graph classification tasks. Figure 1 shows the architecture consisting of three blocks with GCN and HGLA-Pool layers. Each block receives a graph as input and generates a pooled graph with a new feature and adjacency matrices. Concurrently, we use mean-pooling and max-pooling as a readout layer after each block to aggregate all the node representations to obtain a single graph embedding as follows:

$$\Gamma_j = \left[\max_{1 \leq i \leq N^l} X_{ij}^l \right] \forall j \in [0, d]; \quad R^l = \left[\frac{1}{N^l} \sum_{i=1}^{N^l} x_i^l \parallel \Gamma \right] \quad (9)$$

where N^l is the number of nodes at layer l -th, x_i is the feature vector of i -th node, and $\parallel$ denotes concatenation. The $\Gamma \in \mathbb{R}^d$ vector contains the maximum values of each feature dimension across all nodes. The readout $R^l \in \mathbb{R}^{2 \times d}$ concatenates the two (average and max) feature representations. Finally, this graph embedding is fed into a multi-layer perceptron classifier to make predictions.

4. Experiments and Analysis

This section presents a comprehensive evaluation and quantitative analysis of our proposed HGLA-Pool's effectiveness. We perform exhaustive experiments across four diverse dataset categories: chemical molecular structures (Wale et al., 2008) (including Mutagenicity, NCI-1, NCI-109, COX2, and BZR), social networks (Bacciu et al., 2021) (Reddit-Multi-12K), biological networks (Dobson & Doig, 2003) (DD and Proteins) and computer vision (Irwin et al., 2012) (MSRC_21). Table 2 shows the statistics of each dataset. Additionally, we conduct ablation studies to evaluate the individual contributions of each fold within the HGLA-Pool layer. To further analyze the robustness of our approach, we also explore the impact of hyperparameter variations on the performance of our pooling method.

4.1. Baselines and Experimental Settings

We select three GNN architectures to conduct comparative experiments, including GCN (Kipf & Welling, 2016), GAT (Veličković et al., 2017), and GraphConv (Morris et al., 2019) and eleven state-of-the-art hierarchical and global pooling approaches such as DiffPool (Ying et al., 2018), SortPool (Zhang et al., 2018), TopkPool (Gao & Ji, 2019), SAGPool (Lee et al., 2019), CliquePool (Luzhnica et al., 2019), GMT (Baek et al., 2021), MuchPool (Du et al., 2021), MPool (Islam et al., 2023),

Table 1: Comparison of HGLA-Pool and baselines. The highest score is in **bold**, and the second highest score is in underline (OOR referred to as out-of-resources).

Class	Baselines	PROTEINS	D&D	NCI-1	NCI-109	COX2	BZR	MUTAGEN	RED-M-12K	MSRC-21
GNNs	GCN[2017]	73.77 ± 5.59	75.13 ± 4.14	74.68 ± 3.09	73.47 ± 2.22	78.16 ± 0.85	80.01 ± 2.39	78.18 ± 2.54	23.84 ± 0.92	84.04 ± 6.40
	GAT[2018]	75.30 ± 5.23	76.91 ± 2.68	75.16 ± 3.36	73.78 ± 3.38	78.37 ± 0.66	80.74 ± 2.52	78.99 ± 2.44	21.73 ± 0.03	88.80 ± 4.38
	GraphConv[2018]	74.04 ± 5.07	76.57 ± 3.98	78.15 ± 1.86	77.01 ± 2.52	80.79 ± 4.43	83.51 ± 2.89	79.18 ± 2.24	37.53 ± 1.97	88.66 ± 2.49
Pooling	DiffPool[2018]	77.62 ± 4.97	74.31 ± 2.15	70.70 ± 0.22	70.20 ± 0.24	77.60 ± 2.70	78.90 ± 0.04	71.80 ± 0.15	OOR	83.30 ± 0.51
	SortPool[2018]	74.66 ± 5.08	67.47 ± 6.23	70.58 ± 3.68	68.87 ± 2.38	78.20 ± 0.02	79.70 ± 0.07	75.80 ± 2.43	42.50 ± 0.36	74.70 ± 0.40
	TopkPool[2019]	73.32 ± 6.09	73.85 ± 4.63	74.84 ± 4.62	75.09 ± 2.37	77.93 ± 2.79	81.49 ± 4.17	79.56 ± 1.96	37.56 ± 3.39	88.45 ± 3.70
	SAGPool[2019]	73.50 ± 4.56	72.49 ± 2.87	74.26 ± 1.96	74.94 ± 3.13	78.37 ± 1.86	81.24 ± 4.01	79.25 ± 3.03	36.79 ± 3.53	87.92 ± 3.24
	CliquePool [2019]	73.56 ± 2.86	74.81 ± 3.87	77.26 ± 1.53	76.49 ± 1.14	78.37 ± 1.86	82.17 ± 2.25	78.47 ± 1.62	36.11 ± 2.13	88.72 ± 1.14
	GMT[2021]	75.09 ± 0.59	78.72 ± 0.59	74.21 ± 1.88	71.38 ± 2.03	80.86 ± 0.41	82.25 ± 0.70	80.53 ± 0.11	<u>44.06 ± 0.09</u>	<u>90.53 ± 0.34</u>
	MuchPool[2021]	<u>78.52 ± 3.89</u>	76.48 ± 7.01	74.70 ± 2.25	71.84 ± 2.66	79.27 ± 6.01	80.28 ± 6.93	74.75 ± 2.48	OOR	86.85 ± 3.61
	MPool[2023]	79.30 ± 3.30	81.20 ± 2.10	77.40 ± 1.90	73.50 ± 2.50	78.10 ± 0.10	78.70 ± 0.11	79.60 ± 3.70	OOR	87.52 ± 0.54
	MAC[2023]	76.08 ± 3.55	<u>79.13 ± 4.70</u>	77.60 ± 1.66	75.84 ± 1.86	78.36 ± 0.16	82.76 ± 5.91	<u>80.33 ± 1.49</u>	42.67 ± 2.23	89.75 ± 0.22
	Quasi-CliquePool[2023]	75.68 ± 1.38	75.30 ± 3.30	<u>78.21 ± 1.83</u>	<u>77.38 ± 2.23</u>	80.15 ± 2.12	82.87 ± 2.58	79.53 ± 1.58	38.21 ± 2.43	88.15 ± 2.22
	Wit-TopoPool[2023]	75.88 ± 5.60	71.30 ± 0.37	70.58 ± 0.29	69.71 ± 0.25	<u>81.73 ± 3.50</u>	79.75 ± 0.14	73.88 ± 1.29	33.87 ± 1.01	89.16 ± 4.00
	HGLA-Pool+Z_C	76.58 ± 3.34	77.57 ± 3.01	80.80 ± 1.24	79.80 ± 2.30	82.86 ± 3.55	85.93 ± 3.08	82.40 ± 1.58	45.61 ± 2.15	91.30 ± 2.39
Proposed	HGLA-Pool+F_C	73.33 ± 3.18	75.30 ± 3.95	81.02 ± 1.90	79.10 ± 1.10	83.30 ± 3.57	87.43 ± 3.87	82.25 ± 1.56	43.30 ± 2.03	92.19 ± 2.23

Table 2: Eight datasets’ characteristics and Statistics. TG, AN, and AE denote the total number of graphs, average nodes, and average edges, respectively.

Classification	Datasets	TG	AN	AE	Classes
Biological	Proteins	1,113	39.06	72.82	2
	D&D	1,178	284.32	715.66	2
Chemical	NCI-1	4,110	29.87	32.30	2
	NCI-109	4,127	29.68	32.13	2
	Mutagen	4,337	30.32	30.77	2
	COX2	467	41.22	43.45	2
	BZR	405	35.75	38.36	2
Social	RED-M	11,929	391.41	456.89	11
Computer Vision	MSRC_21	563	77.52	198.32	20

MAC (Xu et al., 2022), Quasi-CliquePool (Ali et al., 2023), and Wit-TopoPool (Chen & Gel, 2023).

We implement HGLA-Pool using the PyTorch framework (Paszke et al., 2017) and the PyTorch Geometric library (Fey & Lenssen, 2019). To ensure a fair comparison, we follow many previous works (Lee et al., 2019; Ali et al., 2023; Du et al., 2021), employing tenfold cross-validation to evaluate the effectiveness of all models. We report the average accuracy and standard deviation. For consistency, we utilized the source codes provided by the authors of the baseline models and tuned the hyperparameters according to the specifications in their papers to reproduce the results. We use the same GCN (Morris et al., 2019) layer as a message-passing function for HGLA-Pool and all baselines.¹

Hyperparameters are optimized within predefined ranges, including embedding dimensions {32, 64, 128, 256}, learning rate {0.01, 0.001, 0.0001, 0.05, 0.005, 0.0005}, batch size {64, 128}, pooling and clique ratios {0.5, 0.6, 0.7, 0.8, 0.9}. We employed the Adam optimizer for model initialization and a negative log-likelihood loss function for training. We also adopt patience and early stopping criterion, i.e., if the loss value of the validation set does not reduce for 50 consecutive epochs, then the training process will be stopped.

4.2. Performance Comparison

We evaluated the performance of our proposed method (HGLA-Pool with importance scores Z_C and HGLA-Pool with

dynamic fusion scores F_C) and baseline methods on the nine benchmark datasets for the graph classification tasks. We outlined the classification accuracy with standard deviation in Table 1. In this comparison, HGLA-Pool method achieves superior performance among all other baselines in the chemical molecules, social networks, and computer vision domain datasets. For instance, in the case of the chemical molecular domain, our method outperforms the best baselines by 2.81% improvement for the NCI1, 2.42% for the NCI109, 3.92% for the BZR, 1.57% for the COX2 and 2.07% for the Mutagenicity. It is worth noting that the average number of edges in these datasets is much smaller than in the other datasets (see Table 2). This implies that these five chemical datasets are relatively sparse, presenting a significant challenge for pooling layers to learn meaningful representations. However, our designed dual-fold attention strategy can capture the graph’s local and global information, enabling HGLA-Pool to extract information underterred by the sparse graph structure effectively.

Furthermore, HGLA-Pool consistently outperforms GCN-based global pooling methods across datasets from all four domains. This superior performance underscores the efficacy of our approach in generating more meaningful graph representations, thereby emphasizing the value of incorporating hierarchical pooling layers into the learning process. Notably, HGLA-Pool and other node clustering techniques like CliquePool, Quasi-CliquePool, and MPool perform better than GNN-based models such as GCN, GAT, and GraphConv. This outcome suggests that capturing GTSs as maximal cliques or clusters is beneficial for enhancing graph representation learning. It is also noteworthy that CliquePool and DiffPool do not consistently outperform node selection pooling approaches such as SAG-Pool and TopkPool. This observation further substantiates the idea that integrating the graph’s local and global properties can lead to more effective GP methods. We can also see from Table 1 that our proposed HGLA-Pool demonstrates considerable improvement in the multi-classes based datasets, with an increase of 1.66% and 1.55% on MSRC_21 and REDDIT-MULTI, respectively. Our analysis identified a significant proportion of isolated nodes and subgraphs in the biological dataset. Furthermore, the graphs in this dataset demonstrate a scarcity of structures resembling cliques. Our approach’s effectiveness hinges

¹We reproduced the numbers of all baselines using the source codes given by the authors, and the source codes for all baselines and our method are available on this link.

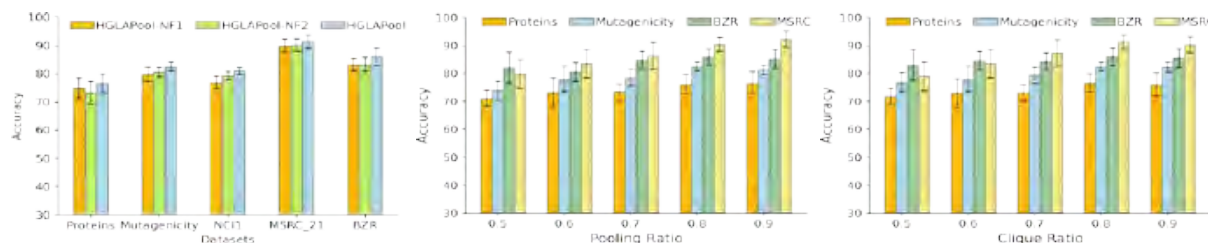


Fig. 2: Effect of dual folds and Hyperparameter analysis on HGLA-Pool performance.

significantly on identifying and ranking such cliques, leading to challenges in capturing meaningful hierarchical representation structures within these graphs. This structural difference between the dataset and our method's underlying assumption likely contributes to the observed performance decline. To sum up, our proposed HGLA-Pool consistently performs better on seven out of nine benchmarks than baseline pooling techniques.

4.3. Ablation Study

This subsection conducts an ablation study on HGLA-Pool by removing independently both folds to verify further where the performance improvement comes from. For convenience, we name the HGLA-Pool method without the first and the second fold as HGLA-Pool-NF1 and HGLA-Pool-NF2, respectively. We chose five different-scale graph datasets covering small and large graphs for these experiments. From Figure. 2 (see the leftmost graph), we can see that capturing GTS is crucial in the chemical molecule and MSRC datasets since the GLA-Pool-NF2 variant obtains a notable performance enhancement as graph attention facilitates the selection of the most informative cliques. Meanwhile, the GLA-Pool-NF1 variant outperforms GLA-Pool-NF2 in the PROTEINS as this dataset has limited clique-like structures, which poses challenges for our method in capturing meaningful hierarchical representation structures. This pattern suggests that the LTS with node features are more important for some datasets than the GTS. Overall, HGLA-Pool's ability to learn both the graph's local and global enables it to generate robust graph representations that significantly enhance performance in classification tasks.

4.4. Parameter Sensitivity Analysis

In this section, we investigate the sensitivities of the two main parameters, P_r and C_r , influencing the performance of the HGLA-Pool. Figure 2 (see second and third graph) summarizes the accuracy of HGLA-Pool under different combinations of these parameters across four different datasets. Our method achieves the highest accuracy for all four datasets when $P_r=0.8$ and $C_r=0.8$, while performance drops with $P_r=0.5$ and $C_r=0.5$. To further investigate these scenarios, we performed a more detailed analysis of the interaction between P_r and C_r , generating a heatmap (see Figure. 3) for each dataset, illustrating their combined influence. The heatmap reveals specific scenarios where certain combinations of P_r and C_r result in sub-optimal performance, particularly in datasets with sparsely connected graphs like Mutagenicity. For instance, low values of P_r and C_r (e.g., $P_r = \{0.5, 0.6, 0.7\}$ and $C_r = \{0.5, 0.6, 0.7\}$) tend to exacerbate graph sparsity, leading to a loss of substantial graph

structures during pooling, as depicted in Figure. 3. From our analysis, we also found that a C_r of at least 0.7 is essential for biological, social, and computer vision datasets, while a minimum C_r of 0.8 is necessary for chemical datasets due to their sparse nature. Therefore, the recommended small ranges for C_r and P_r are $\{0.7, 0.8\}$ for biological, social, and computer vision datasets and $\{0.8, 0.9\}$ for molecular datasets. Interestingly, the HGLA-Pool demonstrated stability, producing reasonably satisfactory results across most combinations of these parameters.

4.5. Graph Visualization

To further demonstrate the distinctiveness and superiority of our pooling method compared to existing techniques, we use the Networkx library to visualize the pooling outcomes of HGLA-Pool, CliquePool, TopkPool, and MuchPool. To provide a fair comparison, we build a hierarchical pooling architecture consisting of two layers and set a 0.8 for the C_r and P_r . We randomly selected a graph from the Mutagenicity dataset comprising 36 nodes for the demonstration. The input and pooled graphs of each method's first and second pooling layers are shown in the first row and the second row of Figure 4, respectively. The results demonstrate that the HGLA-Pool, CliquePool, and MuchPool mostly preserved the input graph's significant topological structure (ring and branch structures). In contrast, the results of the TopKPool are scattered with numerous isolated nodes, indicating a lack of structural preservation. The results obtained from the second pooling layer indicate that three baselines encounter challenges with preserving the underlying topology of the initial graph. However, HGLA-Pool can preserve reasonable topological structures, such as dual ring structures present in the initial graph. This underscores the effectiveness of HGLA-Pool since ring structures are crucial in the characterization of molecules.

4.6. Limitations

Our pooling method, rooted in the fundamental notion of cliques, effectively captures the graph's local and global characteristics. However, its efficacy is constrained when applied to datasets with non-clique structures, such as graphs containing many isolated nodes. For example, if the clique-based representation in fold-1 is inaccurate, it might lead to suboptimal outcomes in subsequent steps, such as clique ranking and LocalPool.

5. Conclusion

This study presents the HGLA-Pool method, an innovative approach for hierarchical graph representation. It employs a

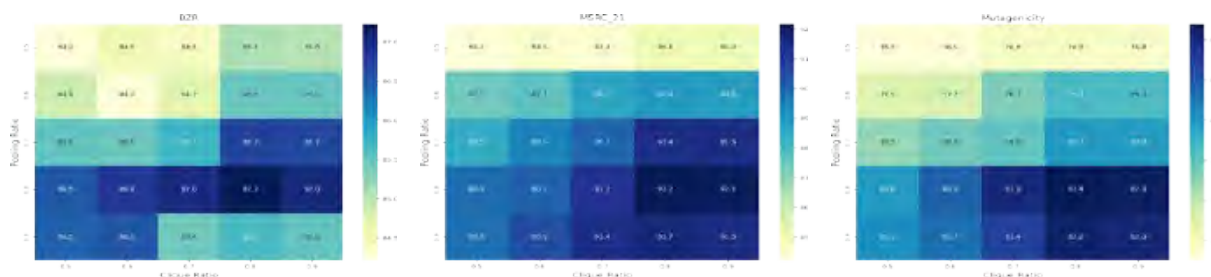


Fig. 3: Effect of different combinations of clique and pooling ratios on HGLA-Pool performance.

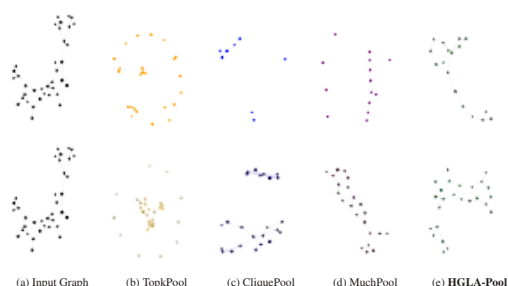


Fig. 4: Comparative graph visualization of different pooling methods. two-fold strategy to capture the graph's global and local properties sequentially. Fold-1 captures the global topological structure of a graph by identifying the cliques and overlapping nodes between cliques, coupled with a dynamic fusion scoring method to rank and select significant cliques for pooling. In Fold-2, a LocalPool layer employing a multi-attention mechanism selects the most informative nodes from these cliques, capturing the LTS. In the end, the outcomes of both folds are integrated sequentially to form a new pooled graph, which retains the original graph's structure and enhances classification performance. The effectiveness of HGLA-Pool has been rigorously assessed through extensive experiments across nine graph classification datasets spanning four distinct domains. Future work will enhance the clique pooling technique to capture a broader range of graph structures, further boosting the graph classification performance.

References

- Ali, W., Vascon, S., Stadelmann, T., & Pelillo, M. (2023). Quasi-cliquepool: Hierarchical graph pooling for graph classification. In *Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing* (pp. 544–552).
- Bacciu, D., Conte, A., Grossi, R., Landolfi, F., & Marino, A. (2021). K-plex cover pooling for graph neural networks. *DMKD*, 35, 2200–2220.
- Baek, J., Kang, M., & Hwang, S. J. (2021). Accurate learning of graph representations with graph multiset pooling. *preprint arXiv:2102.11533*.
- Bai, Y., Ding, H., Bian, S., Chen, T., Sun, Y., & Wang, W. (2019). Simgnn: A neural network approach to fast graph similarity computation. In *Proc. of the twelfth ACM ICWS-DM* (pp. 384–392).
- Cazals, F., & Karande, C. (2008). A note on the problem of reporting maximal cliques. *Theoretical computer science*, 407, 564–568.
- Chen, Y., & Ge, Y. R. (2023). Topological pooling on graphs. In *Proc. of the AAAI* (pp. 7096–7103), volume 37.
- Dobson, P. D., & Doig, A. J. (2003). Distinguishing enzyme structures from non-enzymes without alignments. *Journal of molecular biology*.
- Du, J., Wang, S., Miao, H., & Zhang, J. (2021). Multi-channel pooling graph neural networks. In *IJCAI* (pp. 1442–1448).
- Fey, M., & Lenssen, J. E. (2019). Fast graph representation learning with pytorch geometric. *arXiv preprint arXiv:1903.02428*.
- Gao, H., & Ji, S. (2019). Graph u-nets. In *international conference on machine learning* (pp. 2083–2092). PMLR.
- Han, K., Wang, Y., Guo, J., Tang, Y., & Wu, E. (2022). Vision GNN: An image is worth graph of nodes. *Advances in NIPS*, 35, 8291–8303.
- Irwin, J. J., Sterling, T., Mysinger, M. M., Bolstad, E. S., & Coleman, R. G. (2012). Zinc: a free tool to discover chemistry for biology. *Journal of chemical information and modeling*, 52, 1757–1768.
- Islam, M. I. K., Khanov, M., & Akbas, E. (2023). Mpool: Motif-based graph pooling. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (pp. 105–117). Springer.
- Khemani, B., Patil, S., Kotecha, K., & Tanwar, S. (2024). A review of graph neural networks: concepts, architectures, techniques, challenges, datasets, applications, and future directions. *Journal of Big Data*, 11, 18.
- Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.
- Kong, Y., Li, J., Zhang, K., & Wu, J. (2023). Multi-scale self-attention mixup for graph classification. *Pattern Recognition Letters*, 168, 100–106.
- Lee, J., Lee, I., & Kang, J. (2019). Self-attention graph pooling. In *International conference on machine learning* (pp. 3734–3743). PMLR.
- Luzhnica, E., Da, B., & Lio (2019). Clique pooling for graph classification. *arXiv preprint arXiv:1904.00374*.
- Mavromatis, C., & Karypis, G. (2024). GNN-RAG: Graph neural retrieval for large language model reasoning. *arXiv preprint arXiv:2405.20139*.
- Morris, C., Ritzert, M., Martin, Hamilton, W. L., Lenssen, J. E., Rattan, G., & Grohe, M. (2019). Weisfeiler and leman go neural: Higher-order graph neural networks. In *Proceedings of the AAAI conference* (pp. 4602–4609).
- Pang, Y., Zhao, Y., & Li, D. (2021). Graph pooling via coarsened graph information. In *Proc. of the 44th International ACM SIGIR* (pp. 2177–2181).
- Paszke, A., Gross, S., Chintala, G., Yang, Z., Edward, Lin, A., Zeming, Antiga, L., & Lerer, A. (2017). Automatic differentiation in pytorch. *Automatic differentiation in pytorch*.
- Réau, M., Renaud, N., Xue, L. C., & Bonvin, A. M. (2023a). DeepRank-gnn: a graph neural network framework to learn patterns in protein-protein interfaces. *Bioinformatics*, 39, btac759.
- Réau, M., Renaud, N., Xue, L. C., & Bonvin, A. M. (2023b). DeepRank-gnn: a graph neural network framework to learn patterns in protein-protein interfaces. *Bioinformatics*, 39, btac759.
- Schmitt-Koopmann, F. M., Huang, E. M., Hutter, H.-P., Stadelmann, T., & Darvishy, A. (2022). FormulaNet: A benchmark dataset for mathematical formula detection. *IEEE Access*, 10, 91588–91596.
- Simonovsky, M., & Komodakis, N. (2017). Dynamic edge-conditioned filters in convolutional neural networks on graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3693–3702).
- Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., & Bengio, Y. (2017). Graph attention networks. *arXiv preprint arXiv:1710.10903*.
- Vinyals, O., Bengio, S., & Kudlur, M. (2015). Order matters: Sequence to sequence for sets. *arXiv preprint arXiv:1511.06391*.
- Wale, N., Watson, I. A., & Karypis, G. (2008). Comparison of descriptor spaces for chemical compound retrieval and classification. *KISs*, 14, 347–375.
- Wang, Z., Liu, M., Luo, Y., Xu, Z., Xie, Y., Wang, L., Cai, L., Qi, Q., Yuan, Z., Yang, T. et al. (2022). Advanced graph and sequence neural networks for molecular property prediction and drug discovery. *Bioinformatics*, 38.
- Xu, Y., Wang, J., Guang, M., Yan, C., & Jiang, C. (2022). Multistructure graph classification method with attention-based pooling. *IEEE Transactions on Computational Social Systems*, 10, 602–613.
- Ying, Z., You, J., Morris, X., Hamilton, W., & Leskovec, J. (2018). Hierarchical graph representation learning with differentiable pooling. *Proc. NeurIPS*, 31.
- Zhang, M., Cui, Z., Neumann, M., & Chen (2018). An end-to-end deep learning architecture for graph classification. In *Proc. AIII*. volume 32.

Originally published in Proceedings of the 11th IAPR TC 3 Workshop on Artificial Neural Networks for Pattern Recognition (ANNPR'24), Springer, Montreal, Canada, October 10-12, 2024.

A Simulation Study on Energy Optimization in Building Control with Reinforcement Learning

★

Peter Bolt¹, Volker Ziebart¹, Christian Jaeger¹, Nicolas Schmid¹, Thilo Stadelmann^{1,2}, and Rudolf M. Fuchslin^{1,2}

¹ Zurich University of Applied Sciences Winterthur, Switzerland

² European Centre for Living Technology (ECLT), Ca' Bottacin, Venice, Italy

Abstract. We propose and evaluate a deep reinforcement learning control paradigm for building energy systems. In comparison to other advanced control techniques, namely Model Predictive Control, the reinforcement learning paradigm avoids the costs and uncertainties associated with the requirement for a control-oriented model. We apply a mixed agent for the Proximal Policy Optimization algorithm, similar to the algorithm proposed in [7] as well as a non-discounted finite horizon optimization scheme.

We investigate the capabilities of the proposed reinforcement learning controller regarding energy efficiency, comparing it against the most widely used rule-based control paradigm as a baseline controller. We verify our proposed paradigm in a simulation study with building models implemented in Dymola.

Keywords: smart building · building control · reinforcement learning.

1 Introduction

The transition from fossil fuels to renewable energy sources entails increased complexity in energy management, due to varying availability and the difficulties of storing electrical and thermal energy. Building energy systems (BES), representing an important share of overall energy consumption, offer significant potential in reducing the carbon footprint, provided that an optimized energy management adapted to both availability and usage patterns can be realized.

Prior work has established that advanced control techniques such as Model Predictive Control (MPC) can yield substantial energy savings in BES [3, 5, 12]. However, despite intensive research, especially in MPC for BES, most buildings still rely on simple rule-based controllers (RBC). A possible reason for this is the high computational complexity associated with online MPC, along with the need for an accurate, control-oriented model to describe the physical system in a way amenable to real-time optimization. For many applications, this restricts the model to being essentially linear. Creating this control-oriented model is

* Funded by Innosuisse 31326.1 IP-EE.

2 P. Bolt et al.

often the most time-consuming part of MPC design and can result in a compromise between prediction accuracy and computational complexity that leads to suboptimal controller performance.

This work is motivated by the idea that Deep Machine Learning (ML) holds promise in solving the energy management problem. ML, as a data-driven technique, has gained traction as a powerful paradigm across various application domains, including building control [11, 15]. Particularly in an engineering context [1], ML can complement the model-driven methodologies. We explore the potential advantages of a control system based on deep reinforcement learning (Reinforcement Learning Control, RLC) in the context of BES. We investigate various RLC variants and compare them to conventional controllers. Notably, the investigated RLC schemes do not necessitate a control-oriented model, potentially reducing development time and enhancing controller performance. However, they do require a numerical model of the BES to gather training data—often spanning several simulated years.

We apply a mixed agent for the Proximal Policy Optimization algorithm (PPO) [9], capable of handling the mixed continuous and discrete action space inherent to our problem (see Section 3.2) and a finite horizon optimization scheme to replace the traditional infinite horizon discounting scheme (see Section 3.3). This change is driven by the observation that the discounting scheme made delaying the import of energy appear cheaper to the controller, whereas in reality, it led to increased costs due to energy import becoming an urgent need at an ill-suited moment.

2 Building energy system (BES)

The investigated system is a simplified version of the one used by the authors in [3], which is based on a real application scenario. It integrates a small-scale, grid-connected photovoltaic collector (PV), one heat pump (HP) charging one thermal storage and supplying energy to a low-temperature surface heating. The consumer is a single-family house modeled as a second-order Lumped Parameter Model, consisting of two resistors and two capacitors. Fig. 1a shows the schematic of the installation which integrates

- Ice storage (IceSt) 84 m^3
- Room heating storage (RhSt) 500 liter
- Photovoltaic (PV) collector, 30 m^2 , south-facing
- Brine to water HP with nominal heat output 3.93 kW at B0/W35 with a Coefficient of Performance (COP) of 4.65

The heat pump has a variable speed compressor, allowing it to adjust its operating electrical power (power modulation). Ice storage is a low-temperature storage with a very long time constant and a high capacity. The room heating storage is a high-temperature storage with low capacity.

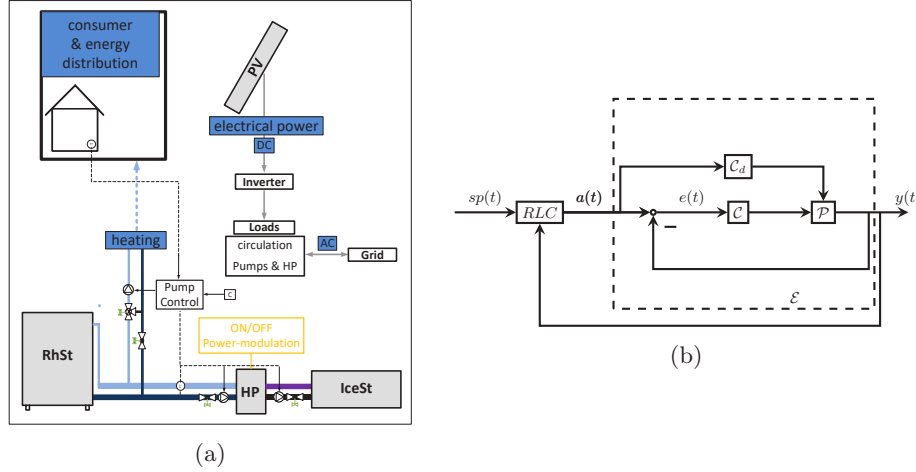


Fig. 1: **a)** Scheme of the HVAC system simulated with Dymola. Heating storage tank (RhSt), heat pump (HP), ice storage (IceSt), and photovoltaics (PV). **b)** Hierarchical control architecture. $\mathcal{P}$: Plant to be controlled, $\mathcal{C}$: continuous (PI) controllers, $\mathcal{C}_d$: discrete HP controller of the inner loop, and $\mathcal{E}$: Environment, constituting the inner closed-loop system. The RL module establishes setpoints for room temperature and HP electrical power, and handles the on/off operation of the HP.

3 Control Architecture

We employ the hierarchical architecture depicted in Fig. 1b. Proportional-Integral (PI) controllers are used in the inner loop to achieve system stabilization for the fast dynamics. The outer loop is a reference governor (RG) [2] for the inner loop. The RG's objective is to minimize the energy consumption of the inner loop while adhering to setpoints and system limitations.

3.1 The reinforcement learning controller

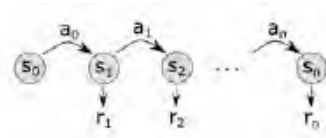


Fig. 2: Markov Decision Process with states s_t , actions a_t and rewards r_t

We formulate the control problem as a Markov Decision Process as depicted in Fig. 2 within the environment $\mathcal{E}$ [13]. The control episode starts a state s_0 .

4 P. Bolt et al.

The controller then successively chooses actions a_t from a set of possible actions $\mathcal{A}$ according to the control policy $\pi(a, s)$ which is the probability of choosing the action a in the state s . Due to the applied actions and the dynamics of the environment, the states s_t will evolve over time. The performance of the controller is measured by the reward r_t which is a function of s_t and the action a_t .

In most RL applications the return R_t is defined as the discounted sum of all future rewards in the control episode of length n

$$R_t = \sum_{l=0}^{n-t} \gamma^l r_{t+l} \quad (1)$$

where $\gamma \in (0, 1]$ is the discount factor, which must be tuned during the training process. A value of $\gamma < 1$ means that rewards count less, the further they lie in the future, reflecting the fact that increased temporal distance introduces more uncertainties. Consequently, smaller yet more assured rewards in the near future are preferred over potentially larger but riskier rewards further from the present.

The return can be used to define a quality measure of the chosen actions, namely the Action-Value Function (Q-function):

$$Q^{\pi, \gamma}(s_t, a_t) := \mathbb{E}[R_{t+1} \mid s_t, a_t] \quad (2)$$

The Q-function represents the expected return when starting from state s_t , taking action a_t , following policy π thereafter, and using the discount factor γ . Similarly, the State-Value Function (V-function) is defined as the expected return when starting from state s_t , following policy π thereafter.

$$V^{\pi, \gamma}(s_t) := \mathbb{E}[R_{t+1} \mid s_t] \quad (3)$$

The function $A(s_t, a_t)^{\pi, \gamma}$ is called advantage and specifies how much better or worse the value of the action a_t is compared to the π -weighed average of all possible actions in the state s_t . It is given by

$$A(s_t, a_t)^{\pi, \gamma} = Q^{\pi, \gamma}(s_t, a_t) - V^{\pi, \gamma}(s_t) \quad (4)$$

In our research, we predominantly utilized the Proximal Policy Optimization (PPO) algorithm as outlined in [9]. PPO is a state-of-the-art reinforcement learning algorithm that combines policy gradient techniques with a clever clipping mechanism. This clipping approach is employed to mitigate the risk of excessively aggressive policy updates that could lead to instability in the learning process. Another advantage of PPO is its ability to engage with multiple environments in parallel, significantly expediting the training process. As described in [10], we approximately maximize the following objective during PPO training:

$$L = \hat{\mathbb{E}}[L_A - c_V L_V + c_S L_S] \quad (5)$$

Here, c_V and c_S are non-negative weighting factors, and L_S contributes to entropy loss for enhanced exploration. Despite the potential benefits of promoting exploration through the entropy term (e.g. [10]), our experimentation did not reveal any notable improvements. Therefore, we set the weighting factor c_S to zero. L_V corresponds to the aggregation of squared deviations between the learned value function V_θ —represented by the neural network’s parameters θ —and the estimated returns $\hat{V} = R_{t+1}$ derived from interactions with the environments.

The first and most important term is given by

$$\hat{\mathbb{E}}[L_A] = \sum_{k=1}^m \sum_{t=0}^n A_{\theta_0}(s_t^k, a_t^k) \frac{\pi_\theta(a_t^k, s_t^k)}{\pi_{\theta_0}(a_t^k, s_t^k)} \quad (6)$$

where θ_0 denotes the parameter set of the neural network that was used when interacting with the environment and generating the data set $\{s_t^k, a_t^k, r_t^k\}_{k=1, \dots, m}$ for the m parallel episodes. When optimizing the loss (equation 6), we employ the following estimate $\hat{A}$ for A [9]:

$$\hat{A}(s_t, a_t) = r_t + \gamma V_{\theta_0}(s_{t+1}) - V_{\theta_0}(s_t) \quad (7)$$

The optimization of L_A can be understood as follows: For $\theta = \theta_0$, the quotient $\lambda = \pi_\theta / \pi_{\theta_0}$ is equal to 1. Starting from $\theta = \theta_0$, the parameters of the neural network are optimized such that λ will be increased for positive advantages (actions with better-than-average action-value) and decreased for actions with negative advantages. Therefore, actions with positive advantages will have a higher probability of being chosen in the next iteration.

In our application, the environment comprises the sub-controllers $\mathcal{C}$ and the Building Energy System $\mathcal{P}$ (see Fig. 1b). The possible actions applied every hour are as follows:

- (i) Heat pump on/off
- (ii) Heat pump power modulation 60 – 100%
- (iii) Indoor temperature setpoint 20 – 23°C

From a BES perspective, if necessary, we can easily convert (ii) and (iii) into discrete actions, provided that the discretization has a fine enough granularity. Concerning (i), the operation of the HP can be approached either discretely by switching the HP on or off every hour or continuously by turning it on or off for a duration between 0s and 3600s.

The state information comprises the current supply and return flow temperatures, the flow rates of the heating circuit, the room temperature of the building, as well as the room heating storage temperature and the power produced by the photovoltaic collector. Additionally, the state includes a 12-hour forecast of outdoor temperature and solar irradiation, updated every hour.

The reward consists of two terms: (i) a deviation of the room temperature from the tolerance band 20–23°C results in a linearly increasing negative reward (a penalty), and (ii) an energy penalty proportional to the energy drawn from the grid. The use of electricity from the PV collector incurs no penalties.

6 P. Bolt et al.

3.2 Mixed Agent

In practice, even actions that are naturally continuous, such as temperature set points, are often discretized and handled by a discrete agent. However, if high resolution is required, the number of discretized actions can become large and it's uncertain whether the same performance can be achieved as with a continuous agent. The mixed Agent is analogous to the one proposed in [7] for the Maximum a Posteriori Policy Optimization (MPO) agent.

In the case of **continuous action** spaces, the Gaussian policy is commonly used. Here, actions $a_c = \{a_{c,i}\}$ are drawn from a parameterized uncorrelated normal distribution represented by the equation:

$$\pi_c(a_c | s) = \prod_i \frac{e^{-\frac{(a_{c,i} - \mu_{\theta,i}(s, a_{c,i}))^2}{2\sigma_{\theta,i}^2}}}{\sqrt{2\pi}\sigma_{\theta,i}} \quad (8)$$

The functions $\mu_{\theta,i}(s, a)$ and $\sigma_{\theta,i}(s, a)$ are approximated by a neural network.

For **discrete action** spaces with actions a_d , the softmax policy is used. It is defined by the equation:

$$\pi_d(a_d | s) = \frac{e^{\phi_{\theta}(s, a_d)}}{\sum_{a'_d \in \mathcal{A}} e^{\phi_{\theta}(s, a'_d)}} \quad (9)$$

Here, the function $\phi_{\theta}(s, a_d)$ is represented by a neural network, and $\pi_d(a_d | s)$ calculates the probability of taking the action a_d given state s . The softmax policy ensures that the probabilities of all possible actions sum up to 1.

For the **mixed agent** with actions $a = \{a_c, a_d\}$ we used the combined distribution given by the product

$$\pi(a, s) = \pi_c(a_c | s) \pi_d(a_d | s) \quad (10)$$

The functions $\mu_{\theta,i}(s, a)$, $\sigma_{\theta,i}(s, a)$, and $\phi_{\theta}(s, a_d)$ as well as $V_{\theta}(s)$ are all approximated by the same neural network with parameters θ , which are found by optimizing the loss given by equation 5.

Exploration in the case of a continuous action space is accomplished by sampling actions according to the latest version of the policy network μ_{θ} and then adding Gaussian noise. This results in the action $a_c = \mu_{\theta} + \mathcal{N}(0, \sigma_{\theta})$, which is subsequently applied within the environment.

For the scenario of a discrete action space, we generate samples from the categorical distribution $\pi_d(a_d | s)$. In this case, the most recent iteration of the policy network ϕ_{θ} is perturbed according to: $a_d = \operatorname{argmax}(\phi_{\theta} - \log(-\log(u)))$, with $u \sim \text{Uniform}(0, 1)$, and the resulting action is applied to the environment.

3.3 Finite horizon optimal control problem

Discounting future rewards (see equation 1) suffers from three disadvantages in our application.

The first one arises from the fact that future energy draws from the grid are discounted which means that the HP seems to need less energy the more the action "HP on" lies in the future. This is briefly illustrated by the following example: Let us assume that we have a sunny winter afternoon and in order to maintain the required room temperature we need to run the HP for one hour in the coming 6 hours. Currently, the PV system provides about 20% of the power the HP needs and therefore 80% must be taken from the grid. Thus, it seems to make sense to immediately switch on the HP and make use of the free PV power. However, if $\gamma = 0.95$ the return, which is always negative in our application, is higher if the power-up of the HP is delayed for the maximum of 5 hours because $\gamma^5 = 0.77$. But running the HP after sunset would certainly cost more grid energy.

This underweighting of future grid energy draws can be reduced by choosing a discount factor very close to one. But then the second disadvantage becomes apparent. Due to the very slow decay of the weighing factor γ^l if $\gamma \approx 1$ rewards that will be generated far in the future with almost no causality to the current action will be taken into account. Furthermore, the long optimization horizon increases the complexity exponentially, resulting in longer computation time and higher variance in training performance.

The authors of [14] discuss how different values of γ would influence the behavior of an RL controller for a heat pump. They compare results with $\gamma = 0.25$ and $\gamma = 0.75$ (both strongly discounted) and observe the following: (i) higher values assign greater importance to achieve long-term rewards and accordingly lead to more frequent operation of the heat pump when the outdoor temperature is high (higher coefficient of performance (COP)). Pre-heating during periods with a high COP consumes more energy at the current time step while saving energy for upcoming time steps. If γ is small, the discounted future savings could not justify the current costs.

Because of these shortcomings we introduced a finite non-discounted optimization window for PPO which favors achieving long-term rewards over a reasonable time horizon without increasing the complexity of the optimization problem too much. The return is given by

$$R_t = \sum_{l=0}^{N_h} r_{t+l} + V_F \quad (11)$$

The prediction horizon N_h replaces the discounting factor γ . V_F accounts for the return value of the final state, reached after N_h steps. We obtained our results by simply setting $V_F = 0$. This can be justified by the fact that the planning horizon is sufficiently long in comparison with the time constants of the system that the final cost term is negligible for the determination of a_t . Additional work may be needed to fully understand the behaviour with respect to this term. A reasonable value for N_h can be estimated based on the time constants present in the system. In our case, the time constants of two thermal heat storages, RhSt and building capacity, are relevant. Equations 2 to 10 remain applicable without modification to this new return formulation.

8 P. Bolt et al.

In the following chapter, we present and compare results for various configurations of the PPO algorithm using both discounted and finite non-discounted reward windows. In addition, we also present results we obtained using the Soft Actor-Critic (SAC) [6] algorithm to find an optimal control strategy. Table 1 lists the investigated configurations.

#	algorithm	action 1	action 2	action 3	γ	N_h
1	PPO _{d:γ=0.95}	on/off	{60, 65, ..., 100}%	{20, 21, 22, 23}°C	0.95	-
2	PPO _{d:γ=0.99}	on/off	{60, 65, ..., 100}%	{20, 21, 22, 23}°C	0.99	-
3	PPO _{d:N_h=12}	on/off	{60, 65, ..., 100}%	{20, 21, 22, 23}°C	-	12
4	PPO _{m:γ=0.95}	on/off	[60, 100]%	[20, 23]°C	0.95	-
5	PPO _{m:γ=0.99}	on/off	[60, 100]%	[20, 23]°C	0.99	-
6	PPO _{m:N_h=12}	on/off	[60, 100]%	[20, 23]°C	-	12
7	SAC	rt. var.	[60, 100]%	[20, 23]°C		

Table 1: Evaluated parameter sets and discrete or continuous actions of PPO and SAC. Actions: (1) Heatpump turned on/off every hour or in case of the SAC the controller can choose a variable runtime ("var.rt.") between 0s and 3600s every hour, (2) Heatpump power modulation in percentage of maximum power, (3) Indoor temperature setpoint.

4 Simulation Results

The RLCs were trained in co-simulation with a BES modeled in Dymola [4]. For the implementation of the RLCs, we employed the OpenAI framework [8]. Our optimization approach for the RLCs aimed at achieving two primary goals: (i) minimizing electricity consumption from the grid and (ii) maintaining room temperature within the range of 20 – 23°C. The range enables the RLC to effectively utilize the building mass as an additional heat storage mechanism (see Fig. 5b). Leveraging the building's mass roughly doubles the short-term storage capacity of the system. In the upper plot of Fig. 5b, it is evident that the RLC capitalizes on this by activating the heat pump during periods of available surplus PV electricity. Throughout the investigation, only a negligible number of small constraint violations were observed. To assess and compare the performance of the RLCs, we have included a baseline RBC and an improved RBC in our evaluation. The baseline RBC uses a heat pump with fixed operating power (no power modulation) and a constant setpoint for the room temperature of 20°C. Furthermore, the heat pump is turned on and off according to a constant threshold temperature in the RhSt. In the improved RBC, both the heatpump power and the room heating storage setpoint were decreased until

there was just enough energy left to cover the requirement. Moreover, a time-dependent threshold temperature in the RhSt was introduced, which increases if PV electricity is available. This adjustment results in significantly higher self-consumption of PV electricity. It is only possible with prior knowledge of the energy requirement and availability of PV-energy, therefore it demonstrates the quasi-upper-performance limit achievable with the used RBC structure. Fig. 3 shows, that this dramatically reduces the RBC's grid electricity consumption.

4.1 Performance Comparison and Training Speed

Fig. 3 illustrates the electricity export versus import from the grid. The controllers minimize electricity import, while export is neither optimized nor constrained. The RLCs demonstrate remarkable energy efficiency results. Among the top-performing RLCs, PPO_d with a prediction horizon of $N_h = 12$ hours consumed an average of $940 \frac{kWh}{a}$ of energy from the electricity grid. A improved RBC, also utilizing the building as a heat storage, required $1280 \frac{kWh}{a}$, while a simple RBC required $2060 \frac{kWh}{a}$. Furthermore, PPO_d exported more electrical energy to the grid than the improved RBC. Overall, RLCs with non-discounted finite windows, specifically PPO_d and PPO_m , exhibit outstanding performance with significantly lower variance compared to RLCs with discounted infinite windows. All simulations were conducted under identical circumstances.

Table 2 shows the number of samples required to achieve 85%, 90%, 95%, and 99% of the maximum reward. The mixed agents achieve high rewards more quickly. For instance, to reach 90% of the maximum performance, $PPO_{m:N_h=12}$ requires 1.8×10^6 samples, while $PPO_{d:N_h=12}$ requires 3.7×10^6 samples.

PPO parameters % of max reward	PPO parameters					
	$d:N_h=12$	$d:\gamma=0.99$	$d:\gamma=0.95$	$m:N_h=12$	$m:\gamma=0.99$	$m:\gamma=0.95$
85%	2.0	3.1	2.0	1.0	1.8	1.6
90%	3.7	3.9	2.9	1.8	2.7	1.8
95%	4.3	4.9	3.1	3.7	3.3	5.1
99%	9.8	8.8	9.2	9.2	7.1	6.9
yearly electricity consumption kWh	951	1022	1032	939	957	928
mean $\pm$ std	± 17	± 55	± 20	± 11	± 33	± 121

Table 2: Number of samples in millions required to reach 85%, 90%, 95%, and 99% of the maximum reward of each agent for the first time.

4.2 Discounted versus Non-discounted Windows

We conducted an empirical investigation to examine the impact of different types of optimization windows. The proposed non-discounted window consistently matches or surpasses the discounted versions in all scenarios, resulting in

10 P. Bolt et al.

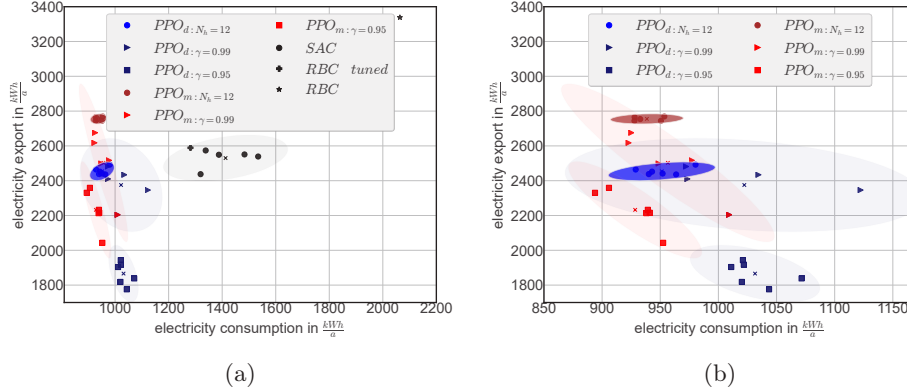


Fig. 3: **a)** Energy efficiency comparison among different controllers optimized to minimize grid electricity consumption (kWh) while maintaining room temperature within $20 - 23^\circ\text{C}$. The vertical axis: surplus energy produced by the PV system and supplied to the grid. Subscripts denote specific properties of the PPO Controller: PPO_d indicates discrete actions, PPO_m indicates mixed actions. The parameter γ represents the discount factor used in the optimization window, if a discounted window was employed. All simulations were run under identical (environmental) conditions. The variance over different runs is due to the stochastic nature of the sampling inherent to reinforcement learning. Fig. **b)** Zoomed-in View of Fig. 3a.

not only lower mean energy consumption but also reduced energy consumption variance. The outcomes are depicted in Fig. 3, where various RLCs with different windows were tested with identical environments.

Optimization becomes more challenging when there's a limited availability of surplus PV electricity. In Fig. 4, we reduced the PV area from $30m^2$ to $15m^2$. Notably, in this scenario, the non-discounted PPO_d outperforms the discounted $PPO_d: \gamma=0.95$ by approximately 13% in grid electricity consumption reduction. Concurrently, the exported electricity increases by over 30%.

5 Conclusion and Outlook

We have shown that an RLC has a great potential for saving primary energy. In contrast to MPC the RLC approach avoids the expenses and compromises associated with developing a control-oriented model.

The use of a mixed agent has significantly accelerated the learning process, achieving performance that is at least comparable to, if not superior to, a standard PPO agent. Nevertheless, the amount of data required remains impractical for online learning applications. In fact, we would need a reduction of several orders of magnitude to eliminate the dependency on building simulations.

Energy Optimization in Building Control with Reinforcement Learning 11

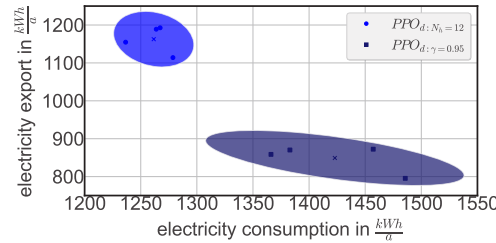


Fig. 4: Simulation with Reduced PV Area. The PV area is $15m^2$, compared to the original $30m^2$.

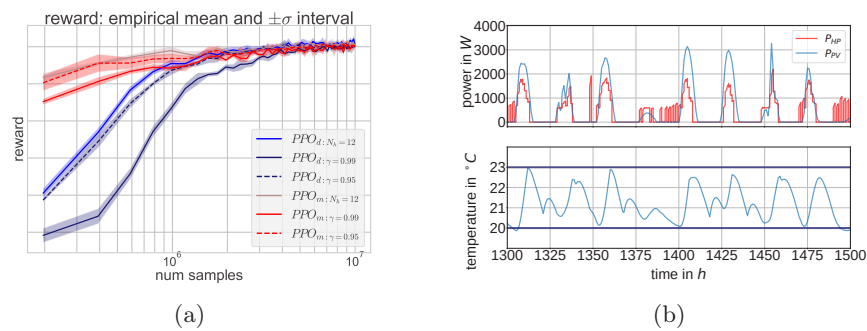


Fig. 5: **a)** Evolution of reward during training. Solid lines: mean performance. The shaded regions: lower 10th and upper 90th percentiles over 6 trials. Note the significant variation in the initial phase. **b)** Top: Electricity production with PV and the actively modulated power consumption of the heat pump. Bottom: Room temperature within the target range of 20 – 23°C.

While deep RL provides a comprehensive framework for deriving optimal control strategies directly from data, further progress is needed to effectively apply RL to real-world BES. Within the current range of RL methods, two intriguing paths for exploration emerge: One involves initiating the learning process with a RBC as the initial policy, rather than starting with a randomly initialized neural network. This approach would facilitate the learning process with a reasonable performance baseline. Alternatively, we could leverage pre-trained models and adapt them, either with or without additional training, for applications in BES.

References

1. Balali, Y., Chong, A., Busch, A., O’Keefe, S.: Energy modelling and control of building heating and cooling systems with data-driven and hybrid models—a review. *Renewable and Sustainable Energy Reviews* **183**, 113496 (2023)
2. Bemporad, A.: Reference governor for constrained nonlinear systems **43**(3), 415–419. <https://doi.org/10.1109/9.661611>

12 P. Bolt et al.

3. Bolt, P., Ziebart, V., Jaeger, C., Ritzmann, R., Meier, O., Fuchsli, R.M.: Model predictive control for building automation. pp. 1330–1341. International Solar Energy Society. <https://doi.org/10.21256/zhaw-3296>, <https://digitalcollection.zhaw.ch/handle/11475/16903>
4. Fritzson, P.: Principles of Object-Oriented Modeling and Simulation with Modelica 2.1. Wiley, Hoboken, NJ (2004)
5. Fux, S.F., Ashouri, A., Benz, M.J., Guzzella, L.: EKF based self-adaptive thermal model for a passive house **68**, Part C, 811–817. <https://doi.org/10.1016/j.enbuild.2012.06.016>, <http://www.sciencedirect.com/science/article/pii/S0378778812003039>
6. Haarnoja, T., Zhou, A., Abbeel, P., Levine, S.: Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor (Aug 2018). <https://doi.org/10.48550/arXiv.1801.01290>, <http://arxiv.org/abs/1801.01290>, arXiv:1801.01290 [cs, stat]
7. Neunert, M., Abdolmaleki, A., Wulfmeier, M., Lampe, T., Springenberg, T., Hafner, R., Romano, F., Buchli, J., Heess, N., Riedmiller, M.: Continuous-discrete reinforcement learning for hybrid control in robotics. In: Conference on Robot Learning. pp. 735–751. PMLR (2020)
8. OpenAI: OpenAI. <https://openai.com/> (2003–2021), accessed: September 2021
9. Schulman, J., Moritz, P., Levine, S., Jordan, M., Abbeel, P.: High-dimensional continuous control using generalized advantage estimation <http://arxiv.org/abs/1506.02438>
10. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms (2017)
11. Stadelmann, T., Tolkachev, V., Sick, B., Stampfli, J., Dürr, O.: Beyond imagenet: deep learning in industrial practice. Applied data science: lessons learned for the data-driven business pp. 205–232 (2019)
12. Sturzenegger, D.: Model predictive building climate control - steps towards practice
13. Sutton, R.S., Barto, A.G.: Reinforcement learning: an introduction. Adaptive computation and machine learning, MIT Press
14. Vázquez-Canteli, J., Kämpf, J., Nagy, Z.: Balancing comfort and energy consumption of a heat pump using batch reinforcement learning with fitted q-iteration **122**, 415–420. <https://doi.org/10.1016/j.egypro.2017.07.429>, <https://www.sciencedirect.com/science/article/pii/S1876610217332629>
15. Wang, Z., Hong, T.: Reinforcement learning for building controls: The opportunities and challenges. Applied Energy **269**, 115036 (2020)

Originally published in Global Resilience White Papers, No. 2, January 28, 2025.

<https://www.globalresiliencepub.com/wp-content/uploads/2025/01/Global-Resilience-White-Paper-2-A-GUIDE-TO-AI-by-Dr.-Thilo-Stadelmann.pdf>

A guide to AI: Understanding the technology, applying it successfully, and shaping a positive future

Thilo Stadelmann

Abstract If artificial intelligence were a road, what would the signs look like? 'No Speed Limit' / 'Caution, Danger Ahead' / 'Toll Station Ahead' / 'Start of Multi-Lane Construction'? Such narratives are indeed told about AI, sometimes contradicting each other. Which ones you trust is crucial for the future – in business, society, and private life. Therefore, we take an analytical look at the state of this road and clear up the forest of signs. We first explore the scientific, philosophical, business, and societal levels of the metaphor 'artificial intelligence' to get closer to its core. We examine the underlying technology and look at primary fields of application and frequently mentioned risks. From this, we derive concrete options for dealing with and implementing AI, and provide an outlook on our future.

1 A roadmap for the high-tech landscape

Many narratives about artificial intelligence (AI) are currently circulating: A utility revolution for the economy [1] with market potential in the trillions [2]; or potential extinction-level threats to humanity through a sudden emergence of hypothetical "artificial general intelligence" (AGI) [3] and other dystopian scenarios [4] – you name it [5]. Like too many signs in a highway construction site, this can be confusing. Which ones should you pay attention to?

This article understands AI as two things: a *powerful tool* on one hand, which can actually be understood, scientifically grasped, and advantageously used; and an *empty phrase* on the other hand, which can be filled with any content and thus fuel hype and, depending on one's worldview, stir up fears or create utopias. Both variants

Prof. Thilo Stadelmann, Dr. rer. nat.
ZHAW Centre for Artificial Intelligence, Technikumstrasse 71, CH-8400 Winterthur, Switzerland
AlpineAI AG, Obere Strasse 22b, CH-7270, Davos, Switzerland
e-mail: stdm@zhaw.ch

of meaning must be considered as they influence our economic, private, and social reality.

How did we get here? The roots of the scientific field of AI go back to the 1950s. The name “artificial intelligence” for the new discipline was chosen for monetary considerations: the founders wanted to attract substantial funding for the new science of “complex computer applications” [6]. This choice continues to provoke more extreme emotions than usual in science, and thus more extreme hype and disappointment, not the least because intelligence and everything associated with it deeply affects humans. For example, AI only emerged from hibernation in the mid-2010s, where it had been sent by public and professional punishment through neglect due to certain unfulfilled promises of the 1990s. As late as 2014, according to a study in Switzerland, only a few research groups were dedicated to the topic [7] (today there are hundreds); and the introduction of an AI curriculum [8] in a computer science program raised major concerns that same year (“Old hat, do we need this?”). At the same time, machine learning was already finding its way into businesses due to its utility [9]. This usefulness reflects the continuous progress in a field that inherently operates in an application-oriented way and thus produces useful tools [10]. The reservations, hypes, and hibernations, on the other hand, are expressions of the extreme expectations projected onto the technology by humans and society through its anthropomorphization.

In the following, we’ll explore the foundations of both meanings of “AI” (Chapter 2) and explain how it works. We go on to exemplarily span the arc of today’s application possibilities and illuminate risks (Chapter 3). Thus equipped with a solid basic understanding of the technology’s possibilities and limitations, we will finally peer into the future (Chapter 4) — which we as humanity will shape ourselves.

2 Background: What is artificial intelligence?

2.1 Scientific foundation

Artificial intelligence is the scientific field concerned with the *mimicking of intelligent behavior using computers*. As such, it belongs to the discipline of computer science. While human capabilities often serve as the benchmark for the quality of results, AI is neither about simulating the underlying biological processes, nor does it methodologically rely on (or contain) a unified theory of intelligence. Rather, it is a methodological *toolbox* of diverse techniques that exploit the advantages of modern computers (large memory for data on which fast, simple computational operations are performed) [11]. These methods can simulate specific, more or less isolated aspects of intelligent behavior. The definitive textbook on the complete field of AI comes from Russell & Norvig [12].

Historically, the “AI toolbox” has two major compartments. In the first one are methods often described as *knowledge-based*, which ultimately aim to process a collection of facts using incorruptible logic to arrive at new statements about the

world. The attempt to reduce all intelligent behavior to logic and thus build a world model can be considered failed [13]. Nevertheless, the underlying methods are used millions of times daily: Fast search across all combinations of individual steps to arrive at a complex goal with a suitable step sequence not only helped IBM's AI system "Deep Blue" defeat reigning chess world champion Garri Kasparov in 1997. It also enables pathfinding in our navigation systems today.

However, it is the second major compartment in the AI toolbox that is responsible for the current boom around intelligent systems. Presumably every currently widely discussed, astonishing AI system, including generative AI, is based on its methods of *machine learning* (ML). These are procedures for generating behavior that we could not satisfactorily express in rules (or program code) manually, but can describe exemplarily through examples. Take the challenge of distinguishing cats from dogs in photos as an example: It is unclear what set of rules would uniquely and correctly describe this separation. For every rule about fur or head shape, exceptions could be found. However, it is simple to compile a *dataset* of dog and cat images from which a human would easily learn the relationship (or *target function*, $f(x) = y$). Here, x would symbolize the input, i.e., an image, y would correspond to the output, say 1 for dog and -1 for cat, and $f()$ would represent the function that maps images x to so-called *labels* y .

In ML, the computer takes over this *training*: It receives a dataset of images $\mathbf{x}$ in a suitable encoding (e.g., as a vector that arranges all pixels as values in sequence, image row by image row) and a suitably adaptable function $f()$ (for example, one that can represent many different curved surfaces through choice of *parameters*). Starting from an initial (e.g., random) configuration, it successively adjusts the parameters of $f()$ to *minimize* the *error* between *predicted* y' and actual y for a given $\mathbf{x}$, and this for all $(\mathbf{x}, y)$ pairs in the dataset.

For simplicity, let's imagine this with images represented by just two pixels $\mathbf{x} = (x_1, x_2)$ —making them merely points in a two-dimensional coordinate system that we can still easily visualize (see Figure 1). We can assume that dog images share some similarity among themselves and thus form a cluster in the 2D space, somewhat distant from all cat images. Therefore, we could try to separate the two clusters with a straight line—a linear equation. The target function $f()$ would then be that a dog image ends up above the line defined by $(\theta_0 + \theta_1 x_0 + \theta_2 x_1 = 0)$ and a cat image below it, thus:

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } \theta_0 + \theta_1 x_0 + \theta_2 x_1 \geq 0 \\ -1 & \text{if } \theta_0 + \theta_1 x_0 + \theta_2 x_1 < 0 \end{cases}$$

Here, the three parameters θ_1 – θ_3 are more intuitively known as the slope ($m = \frac{\theta_1}{\theta_2}$) of the line as well as its intersection point with the vertical axis ($b = \frac{\theta_0}{\theta_2}$), and the linear equation itself as $x_2 = mx_1 + b$. By varying the (two or three) parameters alone, all possible lines in two dimensions can be realized. The goal of machine learning in this example is thus to find a configuration of m and b (or θ_1 – θ_3) (i.e., to place the line in space by rotating and shifting it) so that ideally all $\mathbf{x}$ representing dogs ($y = 1$) end up above the line, while all cat- $\mathbf{x}$ ($y = -1$) land below. For a new image $\mathbf{x}'$,

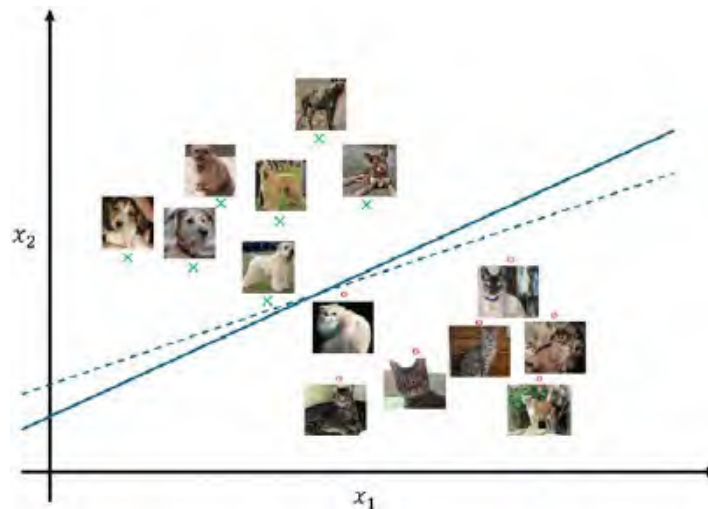


Fig. 1 Images of dogs and cats (source of individual images: ImageNet [14]), represented by only 2 coordinates (red $\circ$, green $\times$). These are perfectly separated into the two categories by the line $\theta_0 + \theta_1 x_0 + \theta_2 x_1 = 0$ (blue, solid line): All dog images are found above the line ($\theta_0 + \theta_1 x_0 + \theta_2 x_1 > 0$), the cats lie below ($\theta_0 + \theta_1 x_0 + \theta_2 x_1 < 0$). Alternative separation lines are also conceivable (such as the dashed, blue line). ML methods find the optimal parameters θ for given encodings $\mathbf{x}$ of the data points and a predetermined function class $f(\cdot)$ (such as the class of linear functions).

the trained *model* $f(\cdot)$ (with ideal parameters found using the training dataset) now directly outputs whether it is predicted to be a cat ($f(\mathbf{x}') < 0$) or dog ($f(\mathbf{x}') \geq 0$).

The example above shows how a specific use case (distinguishing cats from dogs) must first be formalized for the computer (suitable encoding of input x and output y), a training dataset of (x, y) pairs must be compiled, and a suitable function type $f(\cdot)$ must be chosen. The computer then takes over the systematic adjustment of the parameters of $f(\cdot)$, so that the deviation between calculated output and actual label is minimized across all examples. If the training set was large enough and representative of future expected inputs, there is good reason to assume that the learned model *generalizes*. In real applications, where inputs are high-dimensional (such as images, videos, text, sensor recordings) and the relationship with the desired output is complex (nonlinear, like mapping visual appearance to biological species), one would choose a more adaptable target function than a 2D line—such as a so-called neural network, which leads to the *deep learning* variety of ML. Its up to billions of parameters ultimately provide for arbitrary “curviness” of the separation surface from above and thus for high adaptability to the data and the relationship to be modeled. However, the learning principle remains the same. For a brief introduction for non-technical readers, it is recommended to read Chapter 2 from Segessenmann et al.’s article [15]; the definitive textbook on deep learning was recently presented by Prince [16].

What properties does such a ML model have? It was learned directly from the data without much explicit prior knowledge (“data-centric” [17, 18]). This means what

wasn't in the data isn't in the model. The model further maintains a purely statistical view of the world, because the above-described type of target function optimization, "maximum likelihood," indeed implicitly estimates how well the probability distribution of predictions matches the observed distribution in the training data. This statistical view works for many applications on average across many predictions (sometimes significantly) better than what a human could achieve manually. This applies to the above dog-cat classifier just as much as to Large Language Models (LLMs) like ChatGPT, which predict the most probable next word from a context of words. However, the result in individual cases can be completely wrong, as statements about statistical plausibility—to stick with the example of language—are not statements about truth. This should be kept in mind when the next impressive AI demonstration leaves one amazed.

2.2 Philosophical and narrative context

Since technology and particularly science fiction literature have existed, humans have fantasized (in the best sense of the word) about "artificial intelligence"—not as real existing technology or science, but as a container for futuristic ideas. That such "artificial intelligence in Hollywood's science fiction sense" and AI have nothing in common except the same 23 characters in identical configuration should ideally be neither special nor concerning. For instance, the flight recorder also known as "black box" shares its name with entirely unrelated phenomena ("black boxes": things that are considered completely incomprehensible due to their complexity), yet this does not lead to serious misunderstandings. However, AI is—economically and thus socially speaking—currently a game with extremely high stakes, so that mixing the two levels of meaning in public statements becomes problematic: The projection of events from science fiction onto unrelated present-day technologies can evoke fear and uncertainty.

As an example of such a statement, take the open letter from the Future of Life Institute calling for an AI moratorium [19]: The background of the numerous signing researchers and entrepreneurs alone lends technical-scientific credibility to the concern. However, the mentioned risks and proposed solutions are based on extrapolations that are philosophically-ideologically founded and are described as "hypothetical" and "imaginary" by influential parts of the research community [20].

That such narratives around self-improving, general or superintelligent AI exist took its origin at least with the thought experiment known as "Laplace's demon". It constitutes that all true statements about the world could be derived using complete data and perfect models [21]. If (and only if) these two conditions of completeness and perfection would be fulfilled, the resulting demon could operate as a flawless predictor of the future and answer any question by logical deduction. Interestingly, Laplace's motivation in introducing the demon was merely to reduce the associated idea of complete predictability to absurdity—it is infeasible precisely because the two conditions are thwarted by the inherent uncertainty of things. Regardless, this

feasibility fantasy lives on today and finds new philosophical forms, which Émile Torres and Timnit Gebru summarize under the acronym *TESTCREAL* [22]: Materialism thus leads to belief in the singularity (the “S” in *TESCREAL*) and thereby to the possibility that AI could ruin humanity’s distant (longtermism, “L”) but glorious future in space (extropianism “E” and cosmism “C”) as transhuman beings (“T”), since the leap to realizing quasi-human properties in machines could happen at any time and make the technology uncontrollable. Accordingly, rationalism and effective altruism (“R” and “EA”) suggest drastic measures in the here and now. Torres ultimately speaks of these as the “eugenics of the 21st century”—which makes it clear even to non-philosophers that science fiction-fueled AI anxiety should ideally have no broad ideological basis. How widespread it nevertheless is in the companies responsible for a large part of today’s AI progress is exemplarily (and drastically) shown by Leopold Aschenbrenner’s blog [23]: An entire community here pays homage to a questionable idea of the future and influences policy-making worldwide on a belief basis (no scientific or technical arguments involved!) that from an outside perspective is also described as “ridiculous” [24, 25].

So what is artificial intelligence? A collection of methods on one hand, with which certain intelligent behavior can be mimicked using computers today (and better in the future); on the other hand, a term often filled with ideologically shaped content inspired by science fiction, which can generate much attention and fear. Both levels of meaning are partially intertwined in public discourse as well as in product promises.

3 Application: What can artificial intelligence do?

3.1 Business Value

AI is on everyone’s lips not because of a technological revolution (if anything, it would currently be 7 years old: [26]), but due to a utility revolution [1]. For generative AI, particularly text and image generation, the potential applications in both professional and private contexts become obvious to many people after just brief experimentation. Five minutes are enough to get a first impression of the potential, and the entry barriers are incredibly low thanks to simple online access and freemium services. Of course, it is important to note (and absolutely necessary in professional use) to operate in a data-protection-compliant way even during testing, to avoid making internal information public through the prompt history. As a first approximation, only paid versions of corresponding services should be used professionally, where the terms of service usually exclude the provider from freely reusing the information contained in the prompts and replies. For enterprise-wide deployment, specialized companies like AlpineAI¹ offer more control. Alternatively, local instances can be operated [27].

¹ <https://alpineai.swiss/>

If one is lacking ideas about where generative AI could bring the greatest business value, the 100 use cases in the Harvard Business Review guide offer a good starting point [28]. Generally, it is recommended to strictly align any AI deployment (like any other technology) with real business cases: Starting from the lucrative but unsolved problems in each business area is the best way to plan where deployment would be worthwhile in principle. Through experimentation or exchange with specialists, feasibility can then be efficiently clarified in the second step² [29].

With all the benefits of generative AI, it is important not to neglect that AI's methodological toolbox has more to offer: For instance, as part of a data science pipeline for analyzing data points and streams [30] and thus in conjunction with databases and digitalization projects [31, 32, 33]. Here, classical machine learning methods play just as much a role [29] as deep learning, depending on the data context and task [34]. To explore the possibilities in an exemplary way, the following compilation of applications may serve, which teams from the ZHAW Centre for AI have realized together with companies in recent years: In document processing, for example, the segmentation of newspaper pages into individual articles [35], scanning sheet music into machine-readable form [36], or making technical documentation accessible [37, 38]. In medical image processing and the healthcare sector, for instance, improving image-supported diagnosis through data pooling across hospital boundaries [39], accelerating cancer diagnoses through increased automation [40], reducing artifacts in CT images that may result from involuntary patient movement during imaging to acquire better images using less harmful radiation [41], or monitoring intensive care patients to avoid false alarms in nursing care [42]. In the industrial sector, work on production planning for optimized complexity management [43], production parameter estimation for better performance of photovoltaic modules [44], automatic quality control of rotating machinery [9], and process monitoring of plastic injection molding processes [45]. Additional industrial applications for image and time series analysis are discussed in [46, 47].

All mentioned AI systems were developed in direct collaboration with practice partners for everyday use within one to two years, as there were no ready-made methods or products on the market to enable the underlying business case. In particular, none of these applications could so far be replaced by ChatGPT and similar tools, although some of the systems have been in use since 2017. Custom development in partnership between SMEs and AI-experienced research partners thus was not only economically possible but also sensible³. However, if no research gap needs to be filled for the intended solution, results are even possible within a few weeks to months (sometimes even just days) with ROIs in the triple-digit range [51]. This is good news for any organization that has success-relevant questions for which *better*

² <https://data-innovation.org/innovation/> refers to an exemplary innovation process.

³ However, research also includes a risk for failure, including for technical reasons. For example, it has been predicted [48] and later claimed [49] that moderately improved machine learning techniques would make voice recognition applications much more human-like and reliable. However, it turned out on closer inspection [50] that respective AI systems "*cheated*": They were not able to *practically* learn from data what they could model in *principle* because current learning algorithms greedily take a path of least resistance instead building up understanding in a human-like fashion.

predictions in the broadest sense could be helpful (AI is “prediction technology”: The next word given the context; product quality given a measurement; document content given the scan; object type given a photo; etc.). A basic understanding of the technology helps to conduct the initial screening of possible use cases for feasibility oneself, and to proceed to detailed clarifications with partners on the selection with good prognosis.

What kind of skills and business strategy does an organization need on top of this to become successful with AI? Actually, not much to get started. A *fearless* but *cautious* and *determined* attitude helps. Cautious, because risks exist (see below: technical, and otherwise), but determined, since the risk of not acting (and hence missing out, also catastrophically) is real as well. Building up the mentioned basic technical understanding in-house is a very good starting point. A small team that recruits its members from an *alliance of the willing* and spans technical, application, and human factor perspectives, volunteering to start as a track group, is a proven pattern. *Strategically*, a dual approach is advisable: *top-down*, by thinking the business forward: How could our markets / environments change? Where does AI affect what we do, not just how we do it (not limited to, but including thinking about data products [52] and a suitable organizational structure [32])? What will others do—are new players from unusual directions likely? Where do we *want* to go? Which key use cases hold the largest value for being transformed using AI? LLMs can be useful in working through these questions as they can be prompted to act like different personas, thus offering the basis to quantifiable scenario analysis (where a population responding to a poll is simulated by such pre-prompted LLMs).

In addition, another strategic task is to create the framework conditions for the second part of the strategy—the bottom-up part. These are mainly the technical and regulatory conditions to enable every member of the organization to explore the use and potential of AI in their own sphere of influence. This second pillar is important since the sum of all the small potential gains in endless processes where only the people working on them see how to improve them hold an accumulated potential that may exceed the value of the biggest strategic business cases.

For this dual top-down/bottom-up strategy to work, certain cultural values in the organization are definitely an asset, foremost *trust* amongst individuals across organizational levels. As tasks will be transformed and roles might shift, this will also help mitigate potential ensuing volatility in individual work environments. Competitive advantage, after all, comes from the humans involved, their ingenuity and dedication, given the proper use of available tools like AI. AI officers, AI strategies, or availability of AI tools *per se* is more overhead than advantage unless the steps above are implemented (use cases identified, staff engaged, etc.).

3.2 Societal challenges

The opportunities outlined above are accompanied by risks. Many of these are serious yet manageable, particularly those based on shortcomings of current AI

systems with *impact on individuals*, stemming from technical inadequacies of the employed methodology. This includes, for example, the problem of AI bias, meaning that subgroups of society are disadvantaged through the use of an AI system, perhaps because their faces are less well recognized by facial recognition methods (this is disadvantageous when it leads to poorer treatment by police, for instance). Or the problem of adversarial attacks on image classifiers (meaning that through intentional invisible modification of images, the system's results can be manipulated almost arbitrarily). Both problems are well understood [53, 54, 55] and it is absolutely within the power of any organization using such systems to prevent their negative effects using methods of algorithmic fairness. However, the associated effort must be carried out responsibly [56].

Similarly, mitigability is true for risks that arise from the tool-like nature of AI systems in general (beyond their effect on an individual): As a tool, AI can be used for various purposes, good ones (like fighting cancer, see above) and bad ones. The latter include generating fake content (e.g., fake news, deep fakes). When considering this risk, it helps to bring to mind—and this has methodological character for similar questions as well—what is actually new about the risk and what remains constant since pre-AI times: Humans have been faking content since originals existed. What is changing is the scale. Creating arbitrarily professional-looking content is now easy and cost-effective for anyone. There are concerns that a flood of fake content could pose a serious problem for democratic processes and social cohesion. But humans are quick learners and very intolerant of feeling deceived. It is conceivable that they will quickly learn to no longer trust anonymous internet sources in the usual way, and instead help restore more importance to real, trusted relationships. As a supporting measure, AI is of course also being used to detect and filter fake content, which defuses part of the flood. This does not mean that these challenges would not pose a problem. But it puts them in perspective with other societal challenges, which, while serious, are manageable.

Many are concerned about AI's potentially negative impacts on the job market. However, experts often doubt predictions of massive unemployment: AI as a tool does not automate jobs, but individual tasks. Years ago, for example, the profession of radiologist was predicted to decline after computer vision systems achieved excellent results in X-ray diagnostics. Today, there is even a shortage of radiologists in some places, as image analysis makes up only a (small) part of a radiologist's job [57]. The way AI is successfully deployed is in team with humans, as a cognitive orthosis. It extends human capabilities, but like glasses in a pocket, it is quite worthless on its own [58].

The metaphor of AI as a tool also helps in contextualizing specific challenges in education. First, better tools are a positive development. They help us get further in life. However, fundamental skills that enable us to use tools successfully in the first place are learned *without* them. For example, we must learn to calculate to build a solid sense for quantities; and to write, as it trains thinking. The current canon of teaching material thus keeps its relevancy. Language ability, in particular, becomes increasingly important: It now even forms the user interface to computers. Added to this is the need for a basic understanding of technology for everyone, to avoid seeing

the constantly reported breakthroughs as magical, but rather to realistically assess possibilities and limitations and make them usable. As fake becomes increasingly difficult to distinguish from handwork, performance assessments are better conducted orally.

However, the question arises: How will learners motivate themselves in the future to invest the necessary time and effort when the AI solution (always tempting, but not helping the learning objective) is only a few keystrokes away? Learning is associated with pain, and we achieve a good part of the *Good Life* only through voluntarily accepting certain pain. This seems to be the greatest challenge related to AI for society: If AI ideally offers enormous new conveniences and relief, but we only achieve the life we desire if we remain active ourselves (in learning, in relationships, through meaningful activity)—how do we pull ourselves together? How do we remain human (Neil Lawrence sees the core of being human also in its limitedness, which is to be embraced as a strength [59]), when superpowers beckon [60], which are helpful when used specifically and dehumanizing when overstrained? The metaphors for AI could be pointedly supplemented with that of its duality as medicine and drug. One part of the answer will lie in AI being used as tutors, serving a teachers material in my individual learning style, and in human teachers to become master motivators, next to curators of meaning.

4 Future: How do we want to live?

AI will continue to develop in the coming years, unlikely slower than before. At the same time, our society will continue to transform. This follows from the usefulness of already the current methods, leading to them further spreading throughout the economy, and the associated shifts in money that will increasingly be gained with the help of AI (shifting value chains to other departments, different suppliers, operating on other resources) [29]. This is even independent of any major technological progress, as even the current utility of this technology has not been fully exploited yet. If we had a choice, what should be the goal of this transformation?

4.1 Individual recommendations for action

At the individual and organizational level, it makes sense to engage with AI and integrate corresponding systems into daily life and work where sensible and appropriate. *Try it out!* Since development moves quickly, this should happen periodically: What is not good enough today might look different in a quarter.

Looking ahead, which in AI can only reasonably be warranted for at most five years, one can expect AI systems to reach the utility of capable personal assistants. This does not require AGI or superintelligence, but is rather a plausible extrapolation. It stems from the capabilities of today's LLMs, enhanced with the ability for systems

to prompt themselves for a while before returning feedback to humans (so-called “agentic workflows” [61]). Combined with insights from neuroscience, it is also conceivable that the exorbitant hunger for data and electricity, which the current AI development paradigm requires (namely, “scaling” ML systems), could be reduced by orders of magnitude [62, 63]. Corresponding ML methods could also lead to better “world models” that understand their environment less purely statistically and more in terms of cause and effect—thus developing a kind of “machine common sense” that is less susceptible to obviously nonsensical outputs. Again: without AGI.

This raises the question: What actually defines being human and having value [15, 59]? The winners of the future are certainly those people who do not let themselves be intimidated by the capabilities of a fast computer or have their identity questioned. Instead, they know what is good for them as social relational beings, and understand the importance of meaningful life content. This is supported by individually strengthening and engaging with *humanistic* thought that deals with the value and dignity of humans as such, instead of technomorphizing them [15]. This strengthening of human worth and value is something we as humans need to actively pursue—and something we should also put our technology in service of. Future technological development should be guided by asking “does the intended use strengthen what makes us human at the core” (e.g., our thriving in relationships, our limitedness, our need for having responsibility over something). In that sense, the spam filter (shielding us from ever more information overflow) becomes a much more ethical AI application than the next productivity tool (that promises superpowers and might deliver burnout).

4.2 Setting the course for society

On a societal level, as with any major change, the question of regulation arises, and rightly so. Current approaches, however, often focus on compliance and liability. The EU AI Act, for example, sets out only a few specific guidelines on use, but extensively regulates what needs to be documented and how. Returning to the traffic metaphor introduced at the beginning, it is evident that the laws surrounding the emergence of modern mobility also conveyed clear signals about the kind of world their designers wanted to live in: high-speed travel only on highways, traffic calming in residential areas, mandatory vehicle permits and driver licensing, and so on. The aim was evidently to avoid unrestricted high-speed vehicles, anonymity, or untrained drivers. A similar intention in shaping AI could mean regulating not just technology or application domains, but specifically the associated business models. This approach might help avoid collateral damage, as has been seen with social media [64].

Above, we identified the greatest risk of AI as the possibility that, with such “superpowers” on offer, many people might *opt out* of the essential struggles of being human. This would hinder their maturity, making them ill-equipped to thrive in a highly technological world, ultimately cutting them off from the Good Life they desire. Furthermore, a large number of people opting out would become a burden

on a society that depends on cooperation. To counter this, we need societal efforts in *character formation*: it takes the best within us, our best selves, to handle the most powerful tools responsibly. This has implications for curricula (in addition to industry-focused competence orientation, a renewed focus on ethics, philosophy, and spiritual resources that contain deep insights into human worth and value). Perhaps even AI could play a role, serving as a coach to support each of us in the necessary overcoming of oneself. The future, however, is shaped by ourselves. We are the ones having agency.

References

- [1] Pascal Kaufmann, Thilo Stadelmann, and Benjamin F Grewe. *ChatGPT läutet Tech-Revolution ein*. Finanzen und Wirtschaft, <https://www.fuw.ch/chatgpt-laeutet-tech-revolution-ein-303487897856>. 2023.
- [2] Michael Chui, Eric Hazan, Roger Roberts, Alex Singla, Kate Smaje, Alex Sukharevsky, Lareina Yee, and Rodney Zemmel. *The economic potential of generative AI: The next productivity frontier*. McKinsey & Company, available online (19.11.2024): <https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/the-economic-potential-of-generative-ai-the-next-productivity-frontier>. 2023.
- [3] Ruth Fulterer. *Künstliche Intelligenz contra Menschheit: Diesen Kampf gibt es nicht. Er ist nur eine rhetorische Strategie*. NZZ, <https://www.nzz.ch/meinung/kuenstliche-intelligenz-vs-menschheit-diesen-kampf-gibt-es-nicht-er-ist-nur-eine-rhetorische-strategie-ld.1732360>. 2023.
- [4] Yuval Noah Harari. *Homo deus: A brief history of tomorrow*. Harper, 2017.
- [5] Kai-Fu Lee and Chen Qiufan. *AI 2041: Ten visions for our future*. Crown Currency, 2021.
- [6] Thilo Stadelmann, Martin Braschler, and Kurt Stockinger. “Introduction to applied data science”. In: *Applied data science: lessons learned for the data-driven business*. Springer, 2019, pp. 3–16.
- [7] Jean-Daniel Dessimoz, Jana Koehler, and Thilo Stadelmann. “Artificial intelligence research in Switzerland”. In: *AI Magazine* 36.2 (2015), pp. 102–105.
- [8] Thilo Stadelmann, Julian Keuzenkamp, Helmut Grabner, and Christoph Würsch. “The AI-atlas: didactics for teaching AI and machine learning on-site, online, and hybrid”. In: *Education Sciences* 11.7 (2021), p. 318.
- [9] Thilo Stadelmann, Vasily Tolkachev, Beate Sick, Jan Stampfli, and Oliver Dürr. “Beyond ImageNet: deep learning in industrial practice”. In: *Applied data science: lessons learned for the data-driven business* (2019), pp. 205–232.
- [10] Thilo Stadelmann. “KI als Chance für die angewandten Wissenschaften im Wettbewerb der Hochschulen”. In: *Bürgenstock-Konferenz der Schweizer*

Fachhochschulen und Pädagogischen Hochschulen, Luzern, Schweiz, 20.-21. Januar 2023. 2023.

- [11] Rich Sutton. *The Bitter Lesson*. Incomplete Ideas, available online (01.10.2024): <http://www.incompleteideas.net/IncIdeas/BitterLesson.html>. 2019.
- [12] Stuart J Russell and Peter Norvig. *Artificial intelligence: a modern approach, 4th edition*. Pearson, 2022.
- [13] Stephen Wolfram. *Remembering Doug Lenat (1950–2023) and His Quest to Capture the World with Logic*. Stephen Wolfram Writings, available online (01.10.2024): <https://writings.stephenwolfram.com/2023/09/remembering-doug-lenat-1950-2023-and-his-quest-to-capture-the-world-with-logic>. 2023.
- [14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. “Imagenet: A large-scale hierarchical image database”. In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.
- [15] Jan Segessenmann, Thilo Stadelmann, Andrew Davison, and Oliver Dürr. “Assessing deep learning: a work program for the humanities in the age of artificial intelligence”. In: *AI and Ethics* (2023), pp. 1–32.
- [16] Simon JD Prince. *Understanding deep learning*. MIT press, 2023.
- [17] Paul-Philipp Luley, Jan M Deriu, Peng Yan, Gerrit A Schatte, and Thilo Stadelmann. “From concept to implementation: the data-centric development process for AI in industry”. In: *2023 10th IEEE Swiss Conference on Data Science (SDS)*. IEEE. 2023, pp. 73–76.
- [18] Thilo Stadelmann, Tino Klamt, and Philipp H Merkt. “Data centrism and the core of Data Science as a scientific discipline”. In: *Archives of Data Science, Series A* 8.2 (2022).
- [19] Future of Life Institute. *Pause Giant AI Experiments: An Open Letter*. FLI Open Letters, available online (02.10.2024): <https://futureoflife.org/open-letter/pause-giant-ai-experiments/>. 2023.
- [20] Timnit Gebru, Emily M. Bender, Angelina McMillan-Major, and Margaret Mitchell. *Statement from the listed authors of Stochastic Parrots on the “AI pause” letter*. DAIR Institute Blog, available online (02.10.2024): <https://www.dair-institute.org/blog/letter-statement-March2023/>. 2023.
- [21] Pierre-Simon Laplace. *Essai philosophique sur les probabilités*. Courcier, 1814.
- [22] Émile P Torres. *The Acronym Behind Our Wildest AI Dreams and Nightmares*. Truthdig, available online (01.10.2024): <https://www.truthdig.com/articles/the-acronym-behind-our-wildest-ai-dreams-and-nightmares>. 2023.
- [23] Leopold Aschenbrenner. *The Decade Ahead*. Situational Awareness, available online (01.10.2024): <https://situational-awareness.ai/>. 2024.
- [24] Melissa Heikkilä. *Meta’s AI leaders want you to know fears over AI existential risk are “ridiculous”*. MIT Technology Review, available online (03.10.2024): <https://www.technologyreview.com/2023/06/20/1075075/metas-ai-leaders-want-you-to-know-fears-over-ai-existential-risk-are-ridiculous/>. 2023.

- [25] Andrew Ng. *A Victory for Innovation and Open Source*. The Batch Letters, available online (03.10.2024): <https://www.deeplearning.ai/the-batch/a-victory-for-innovation-and-open-source/>. 2024.
- [26] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. “Attention is all you need”. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. NIPS’17. Long Beach, California, USA: Curran Associates Inc., 2017, pp. 6000–6010. ISBN: 9781510860964.
- [27] Lukas Tuggener, Pascal Sager, Yassine Taoudi-Benchekroun, Benjamin F Grewe, and Thilo Stadelmann. “So you want your private LLM at home?: a survey and benchmark of methods for efficient GPTs”. In: *11th IEEE Swiss Conference on Data Science (SDS), Zurich, Switzerland, 30-31 May 2024*. ZHAW Zürcher Hochschule für Angewandte Wissenschaften. 2024.
- [28] Marc Zao-Sanders. *How People Are Really Using GenAI*. Harvard Business Review, available online (02.10.2024): <https://hbr.org/2024/03/how-people-are-really-using-genai>. 2024.
- [29] Thilo Stadelmann. “Wie maschinelles Lernen den Markt verändert”. In: *Digitalisierung: Datenhype mit Werteverlust? Ethische Perspektiven für eine Schlüsseltechnologie*. SCM Hänssler, 2019, pp. 67–79.
- [30] Martin Braschler, Thilo Stadelmann, and Kurt Stockinger. “Data science”. In: *Applied Data Science: Lessons Learned for the Data-Driven Business*. Springer, 2019, pp. 17–29.
- [31] Thilo Stadelmann, Kurt Stockinger, Martin Braschler, Mark Cieliebak, Gerold Baudinot, Oliver Dürr, and Andreas Ruckstuhl. “Applied data science in Europe: Challenges for academia in keeping up with a highly demanded topic”. In: *9th European Computer Science Summit, Amsterdam, Niederlande, 8-9 October 2013*. 2013.
- [32] Thilo Stadelmann, Kurt Stockinger, Gundula Heinatz Bürki, and Martin Braschler. “Data scientists”. In: *Applied Data Science: Lessons Learned for the Data-Driven Business*. Springer, 2019, pp. 31–45.
- [33] Kurt Stockinger, Martin Braschler, and Thilo Stadelmann. “Lessons learned from challenging data science case studies”. In: *Applied Data Science: Lessons Learned for the Data-Driven Business*. Springer, 2019, pp. 447–465.
- [34] Thilo Stadelmann, Mohammadreza Amirian, Ismail Arabaci, Marek Arnold, Gilbert François Duivesteijn, Ismail Elezi, Melanie Geiger, Stefan Lörwald, Benjamin Bruno Meier, Katharina Rombach, et al. “Deep learning in the wild”. In: *Artificial Neural Networks in Pattern Recognition: 8th IAPR TC3 Workshop, ANNPR 2018, Siena, Italy, September 19–21, 2018, Proceedings 8*. Springer. 2018, pp. 17–38.
- [35] Benjamin Meier, Thilo Stadelmann, Jan Stampfli, Marek Arnold, and Mark Cieliebak. “Fully convolutional neural networks for newspaper article segmentation”. In: *2017 14th IAPR International conference on document analysis and recognition (ICDAR)*. Vol. 1. IEEE. 2017, pp. 414–419.

- [36] Lukas Tuggener, Raphael Emberger, Adhiraj Ghosh, Pascal Sager, Yvan Putra Satyawana, Javier Montoya, Simon Goldschagg, Florian Seibold, Urs Gut, Philipp Ackermann, et al. “Real world music object recognition”. In: *Transactions of the International Society for Music Information Retrieval* 7.1 (2024), pp. 1–14.
- [37] Felix M Schmitt-Koopmann, Elaine M Huang, Hans-Peter Hutter, Thilo Stadelmann, and Alireza Darvishy. “FormulaNet: A benchmark dataset for mathematical formula detection”. In: *IEEE Access* 10 (2022), pp. 91588–91596.
- [38] Felix M Schmitt-Koopmann, Elaine M Huang, Hans-Peter Hutter, Thilo Stadelmann, and Alireza Darvishy. “MathNet: A Data-Centric Approach for Printed Mathematical Expression Recognition”. In: *IEEE Access* (2024).
- [39] Pascal Sager, Sebastian Salzmänn, Felice Burn, and Thilo Stadelmann. “Unsupervised domain adaptation for vertebrae detection and identification in 3D CT volumes using a domain sanity loss”. In: *Journal of Imaging* 8.8 (2022), p. 222.
- [40] Peter R Jermain, Martin Oswald, Tenzin Langdun, Santana Wright, Ashraf Khan, Thilo Stadelmann, Ahmed Abdulkadir, and Anna N Yaroslavsky. “Deep learning-based cell segmentation for rapid optical cytopathology of thyroid cancer”. In: *Scientific Reports* 14.1 (2024), p. 16389.
- [41] Mohammadreza Amirian, Javier A Montoya-Zegarra, Ivo Herzig, Peter Eggenberger Hotz, Lukas Lichtensteiger, Marco Morf, Alexander Züst, Pascal Paysan, Igor Peterlik, Stefan Scheib, et al. “Mitigation of motion-induced artifacts in cone beam computed tomography using deep convolutional neural networks”. In: *Medical Physics* 50.10 (2023), pp. 6228–6242.
- [42] Raphael Emberger, Jens Michael Boss, Daniel Baumann, Marko Seric, Shufan Huo, Lukas Tuggener, Emanuela Keller, and Thilo Stadelmann. “Video object detection for privacy-preserving patient monitoring in intensive care”. In: *2023 10th IEEE Swiss Conference on Data Science (SDS)*. IEEE. 2023, pp. 85–88.
- [43] Lukas Hollenstein, Lukas Lichtensteiger, Thilo Stadelmann, Mohammadreza Amirian, Lukas Budde, Jürg Meierhofer, Rudolf M Fuchslin, and Thomas Friedli. “Unsupervised learning and simulation for complexity management in business operations”. In: *Applied data science: lessons learned for the data-driven business* (2019), pp. 313–331.
- [44] Mattia Battaglia, Ennio Comi, Thilo Stadelmann, Roman Hiestand, Beat Ruhstaller, and Evelyne Knapp. “Deep ensemble inverse model for image-based estimation of solar cell parameters”. In: *APL Machine Learning* 1.3 (2023).
- [45] Peng Yan, Ahmed Abdulkadir, Giulia Aguzzi, Gerrit Schatte, Benjamin F Grewe, and Thilo Stadelmann. “Automated process monitoring in injection molding via representation learning and setpoint regression”. In: *11th IEEE Swiss Conference on Data Science (SDS), Zurich, Switzerland, 30-31 May 2024*. ZHAW Zürcher Hochschule für Angewandte Wissenschaften. 2024.
- [46] Niclas Simmler, Pascal Sager, Philipp Andermatt, Ricardo Chavarriaga, Frank-Peter Schilling, Matthias Rosenthal, and Thilo Stadelmann. “A survey

- of un-, weakly-, and semi-supervised learning methods for noisy, missing and partial labels in industrial vision applications”. In: *2021 8th Swiss Conference on Data Science (SDS)*. IEEE. 2021, pp. 26–31.
- [47] Peng Yan, Ahmed Abdulkadir, Paul-Philipp Luley, Matthias Rosenthal, Gerrit A Schatte, Benjamin F Grewe, and Thilo Stadelmann. “A comprehensive survey of deep transfer learning for anomaly detection in industrial time series: Methods, applications, and directions”. In: *IEEE Access* (2024).
- [48] Thilo Stadelmann and Bernd Freisleben. “Unfolding speaker clustering potential: a biomimetic approach”. In: *Proceedings of the 17th ACM international conference on Multimedia*. 2009, pp. 185–194.
- [49] Thilo Stadelmann, Sebastian Glinski-Haefeli, Patrick Gerber, and Oliver Dürr. “Capturing suprasegmental features of a voice with RNNs for improved speaker clustering”. In: *Artificial Neural Networks in Pattern Recognition: 8th IAPR TC3 Workshop, ANNPR 2018, Siena, Italy, September 19–21, 2018, Proceedings 8*. Springer. 2018, pp. 333–345.
- [50] Daniel Neururer, Volker Dellwo, and Thilo Stadelmann. “Deep neural networks for automatic speaker recognition do not learn supra-segmental temporal features”. In: *Pattern Recognition Letters* 181 (2024), pp. 64–69.
- [51] Dorian Selz. *Where’s the ROI for AI?* LinkedIn Post, available online (03.10.2024): https://www.linkedin.com/posts/dselz_wheres-the-roi-for-ai-lets-get-real-activity-7247176825972387840-rLWN. 2024.
- [52] Jürg Meierhofer, Thilo Stadelmann, and Mark Cieliebak. “Data products”. In: Springer, 2019, pp. 47–61.
- [53] Mohammadreza Amirian, Friedhelm Schwenker, and Thilo Stadelmann. “Trace and detect adversarial attacks on CNNs using feature response maps”. In: *Artificial Neural Networks in Pattern Recognition: 8th IAPR TC3 Workshop, ANNPR 2018, Siena, Italy, September 19–21, 2018, Proceedings 8*. Springer. 2018, pp. 346–358.
- [54] Stefan Glüge, Mohammadreza Amirian, Dandolo Flumini, and Thilo Stadelmann. “How (not) to measure bias in face recognition networks”. In: *IAPR Workshop on Artificial Neural Networks in Pattern Recognition*. Springer. 2020, pp. 125–137.
- [55] Samuel Wehrli, Corinna Hertweck, Mohammadreza Amirian, Stefan Glüge, and Thilo Stadelmann. “Bias, awareness, and ignorance in deep-learning-based face recognition”. In: *AI and Ethics* 2.3 (2022), pp. 509–522.
- [56] Eleonora Viganò, Corinna Hertweck, Christoph Heitz, and Michele Loi. “People are not coins: Morally distinct types of predictions necessitate different fairness constraints”. In: *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*. 2022, pp. 2293–2301.
- [57] Saurabh Jha and Eric J Topol. “Upending the model of AI adoption”. In: *The Lancet* 401.10392 (2023), p. 1920.
- [58] Kenneth M Ford, Patrick J Hayes, Clark Glymour, and James Allen. “Cognitive Orthoses: Toward Human-Centered AI”. In: *AI Magazine* 36.4 (2015), pp. 5–8.

A guide to AI: Understand, apply, and shape a positive future

17

- [59] Neil D Lawrence. *The atomic human: Understanding ourselves in the age of AI*. Allen Lane, 2024.
- [60] Andy Crouch. *The Life We're Looking for: Reclaiming Relationship in a Technological World*. Convergent Books, 2022.
- [61] Andrew Ng. *Agentic Design Patterns Part 1*. The Batch Letters, available online (03.10.2024): <https://www.deeplearning.ai/the-batch/how-agents-can-improve-llm-performance/>. 2024.
- [62] Christoph von der Malsburg, Thilo Stadelmann, and Benjamin F Grewe. "A theory of natural intelligence". In: *arXiv preprint arXiv:2205.00002* (2022).
- [63] Pascal J Sager, Jan M Deriu, Benjamin F Grewe, Thilo Stadelmann, and Christoph von der Malsburg. "The Dynamic Net Architecture: Learning Robust and Holistic Visual Representations Through Self-Organizing Networks". In: *arXiv preprint arXiv:2407.05650* (2024).
- [64] Jeff Orlowski. *The social dilemma*. Documentary, available as a Netflix original, see also (03.10.2024): <https://www.thesocialdilemma.com/>. 2020.

8 Research Output of the Natural Language Processing Group

8 Research Output of the Natural Language Processing Group

The NLP research group is led by Prof. Dr. Mark Cieliebak and develops technologies for the analysis, understanding and generation of speech and text. We combine methods from Linguistics, Natural Language Processing (NLP) and Large Language Models with Artificial Intelligence to enable natural language communication between humans and machines.

In our research, we work on topics such as text classification (e.g., sentiment analysis, hate speech detection), chatbots/dialogue systems, text summarization, speech-to-text, speaker recognition and large language models. At the end of 2024, the group consisted of 8 researchers: 1 full professor, 3 senior scientists, 2 research assistants, and 2 PhD students. One of the research assistants pursues her master's degree, and there are five additional non-staff master's students. Our scientific work has been accepted and published at prestigious conferences in NLP such as ACL and ICWSM.

Large Language Models (LLMs) such as *ChatGPT* have recently changed the entire field of NLP. Many classical NLP tasks such as sentiment analysis, named entity recognition or text summarization have become almost trivial and obsolete. Hence, the focus of the research community – and also of our NLP research team – has shifted to other topics.

One major question for LLMs is how to *evaluate the quality of the generated output*. We are approaching this task from two directions: in a fundamental research project funded by SNF, we are developing a statistical model that combines human evaluations with automated metrics. On the other hand, we are supporting Swiss companies who run LLMs in real-life in assessing the quality of their systems.

Retrieval Augmented Generation (RAG) is a paradigm for LLMs that gained a lot of momentum in 2024. Correspondingly, we developed and implemented several RAG systems, e.g. for exam preparation for students or for news research.

The *Swiss AI* initiative is a large-scale academic initiative to build a Swiss GPT model, with more than 100 participating NLP experts. We oversaw the data collection subproject, where we gathered large amounts of Swiss textual data that will now be used to train a Swiss LLM from scratch.

For *Swiss German*, we successfully finished our SNF project for speech-to-text. The models are available for academic and commercial use. Based on this success, we started working on the opposite direction, building a first text-to-speech system to synthesize Swiss German dialects in arbitrary voices.

In addition, we have been working on other topics such as detecting malicious use of social media (e.g. hate speech or misinformation campaigns), a simulator for children in police interviews, a chatbot for language learners, digitization of chess scorecards, and we actively contributed to the NLP community in Switzerland by managing the SwissNLP association, by co-organizing the Swiss Text Analytics conference (SwissText) and the Swiss NLP Days 2025, and by organizing several NLP expert meetings.

We thank all project partners, collaborators, funding agencies and students for their support!

The NLP group, as of March 2025:



Mark Cieliebak, Don Tuggener, Manuela Hürlimann, Jan Milan Deriu, Pius von Däniken, Katsiaryna Mlynchyk, Patrick Giedemann, Lars Schmid

Improving Quantification with Minimal In-Domain Annotations: Beyond Classify and Count

Pius von Däniken¹, Jan Milan Deriu¹, Alvaro Rodrigo², Mark Cieliebak¹

¹Zurich University of Applied Sciences (ZHAW), Winterthur, Switzerland

²NLP & IR Group at UNED, Madrid, Spain

vode@zhaw.ch, deri@zhaw.ch, alvaror@lsi.uned.es, ciel@zhaw.ch

Abstract

Quantification is the task of estimating the class distribution in a given collection. With the growing availability of classification models, the use of classifiers for quantification has become increasingly popular, carrying the promise of eliminating the need for manual annotation. However, the naive classify and count approach presents clear limitations, especially evident in the face of domain discrepancies. In this work, we introduce two novel quantification methods, called CPCC and BCC, which can adapt to new target datasets with a small number of annotated in-domain samples ($N = 100$). To explore their real-world applicability, we apply our methods to a range of quantification tasks in the realm of hateful and offensive language, where they perform markedly better than classify and count and other existing methods.

1 Introduction

The advent of large pre-trained language models (Devlin et al. 2019; Liu et al. 2019; Brown et al. 2020, i.a.) based on the transformer architecture (Vaswani et al. 2017) has significantly improved many Natural Language Processing (NLP) tasks. In particular, they represent the de-facto standard for modern text classification systems. Combined with open science efforts encouraging authors to share their models, and platforms such as *Hugging Face* (Wolf et al. 2019) simplifying distribution, it has become easier than ever to gain access to high-quality off-the-shelf models for many classification tasks.

As a consequence, the end-user of a specific model is not necessarily the same as its developer. This also means that the data a classifier is applied to often does not necessarily match its training data. This is known as domain transfer, which is still a challenging problem (Deriu et al. 2017; Tatlat, Thorne, and Bingel 2018; Toraman, Şahinuç, and Yilmaz 2022).

In this work, we will explore the common setting where an analyst wants to determine the number of offensive posts in a given data collection. To avoid the costly process of manual annotation, they download an off-the-shelf classifier and run it on their data to count the number of offensive posts. This approach seems reasonable but gives rise

to several particular challenges. The first is that the classifier was probably trained on a different domain, and the performance achieved on the training domain does not transfer to the target domain. Secondly, the annotation scheme used in the training data might differ significantly from the intended use of the analyst (e.g., offensive language is not uniformly defined (Vidgen and Derczynski 2020)). Thirdly, the prevalence of offensive language often differs in the training regime and the real world, leading to overestimating the true prevalence of offensive language. Thus, counting the number of offensive posts (i.e., the prevalence) according to a classifier is not straightforward.

Estimating the prevalence or empirical distribution of a certain class in a given set of samples is known as *quantification* (Forman 2008). This is a collection-level task, whereas, in classification, the task is to make accurate sample-level predictions. Estimating the prevalence of hateful and offensive language in social media is of keen interest to social science researchers to study societal trends (Solovev and Pröllochs 2022; Siegel et al. 2018).

The approach taken by our imaginary analyst is known as *classify and count* (Forman 2008). It can be shown that the resulting prevalence estimate depends on the true positive rate (TPR) and false positive rate (FPR) of the binary classifier at hand. Forman (2008) shows that there is the following relationship between the true prevalence p and the estimated prevalence $\hat{p}$ depending on the TPR and the FPR: $\hat{p} = p(TPR - FPR) + FPR$. Hence, the estimated prevalence can be substantially biased. Intuitively, we should be able to solve the above formula for p to get an improved estimate. However, in our scenario, the true TPR and FPR on the target dataset are unknown and often do not match the ones achieved on validation data during development. This point is of particular importance in the case of hateful and offensive language, where the exact inclusion criteria can be fuzzy and differ significantly depending on the data curator (Vidgen and Derczynski 2020), which can lead to underwhelming performance when applying a classifier to a different dataset.

In this work, we tackle the issue of applying a black-box classifier to estimate the prevalence of offensive language. In contrast to work in traditional transfer learning, we assume that we do not have access to the underlying classifier, which does not allow for fine-tuning on annotated data. However,

our proposed solution assumes that the analyst collects annotations for a small subset (e.g., $N = 100$) of their target data. In this case, we will show that we can get high-quality prevalence estimates for a range of classifiers and training datasets. More concretely, our main contributions are:

- Two novel quantification approaches that specifically tackle the issue of transferring the quantification capabilities of a classifier from one domain to another with a small set of only $N = 100$ samples.
- A novel sample selection method to collect the small set of samples used for domain adaptation.
- A large-scale set of experiments¹ showcasing the difficulty of domain transfer in the quantification setting, and that the performance of our novel approaches is superior to existing ones.

2 Related Work

2.1 Quantification

The problem of using classifiers for quantification has been studied in depth by Forman (2008, 2005). Indeed, many of the methods we will describe in Section 3 were first described there. González et al. (2017) provide a survey of quantification methods. In particular, they distinguish three high-level approaches: *classify*, *count*, and *correct*; *adapting classification training algorithms*; and *distribution matching*. The methods explored in this work all follow the *classify*, *count*, and *correct* approach. This is mainly due to our imagined setting, where we assume access to a black-box classifier. In the *correct* step of this class of approaches, we will need access to estimates of the TPR and FPR of the classifier. Forman (2008) proposes to estimate these during training using cross-validation. We will instead assume that there is a small set of in-domain annotated data that we can use for this.

Training dedicated quantification models, for example by adapting classification training algorithms for quantification, represents a promising avenue of research. Esuli and Sebastiani (2015) propose to adapt a structured SVM learning paradigm (Joachims 2005) in order to directly minimize the quantification error. This class of approaches does not fit our setting, as we will assume that we do not have any control over the training process, but only have access to a final model trained as a classifier.

The problem of quantification in textual data has been the topic of a few shared tasks. The *SemEval* series of shared tasks included quantification as a sub-task in their sentiment analysis challenges (Nakov et al. 2016; Rosenthal, Farra, and Nakov 2017). The *LeQua* shared task (Esuli et al. 2022) involved creating quantification systems for sentiment and the topic of product reviews.

2.2 Hateful and Offensive Language Detection

The quick growth of social networks has increased the interest in the automatic detection of offensive or harmful language (Hajibabae et al. 2022). Although there are differ-

¹The code to run our experiments is available at <https://github.com/vodezhaw/icwsm2024/>

ent definitions, the task mainly focuses on detecting content targeting individuals or entities using aggressive language (Vidgen and Derczynski 2020). Current technologies rely on data-driven approaches, based on training models on annotated data. However, the annotation criteria usually depend on the target receiving the offense comment (Vidgen et al. 2019), which might produce some bias. As a consequence, knowledge transference among collections is difficult, and therefore, the application to new data. Additionally, the class distribution highly depends on the method followed for data acquisition and might not reflect a real scenario.

Current methods mostly use transformer-based models (Liu et al. 2022). Some innovations apply an ensemble of models (Roy, Bhawal, and Subalalitha 2022), or include additional information, such as the detection of improper language (Plaza-del Arco et al. 2022) or multiword expressions (Zampieri, Illina, and Fohr 2023), for improving results. Although there have been several studies about transfer learning for hate speech detection (Markov and Daelemans 2021; Ali et al. 2022), the proposed systems usually only perform well on a test set with a distribution similar to the one seen during training (Toraman, Şahinuç, and Yılmaz 2022).

A high interest in hate speech detection can also be seen in the constant proposal of shared tasks. For example, *HateEval* (Basile et al. 2019), *OffensEval* (Zampieri et al. 2019b) at *SemEval 2019*, and a second edition of *OffensEval* at *SemEval 2020* (Zampieri et al. 2020) attracted a lot of participants and the datasets are still widely used. In fact, we include these datasets in our experiments (see Section 5.1). But there have been also several efforts in other languages such as Spanish with the *MEX-A3T* (Aragón et al. 2019, 2020) and *MeOffendEs* (Plaza-del Arco et al. 2021) tasks at *IberLEF*, the *OSACT4* shared task on Arabic (Mubarak et al. 2020) and in German at *GermEval 2018* (Wiegand, Siegel, and Ruppenhofer 2019).

2.3 Calibration

A desirable property of probabilistic classifiers is that they be *calibrated*. This means that their predicted score represents a true probability, meaning, for example, that in 80% of cases where the model gives a score of 0.8, the true label should be positive.

Guo et al. (2017) find that many neural network based classifiers at the time were poorly calibrated and discuss a variety of approaches to re-calibrate them. We will rely on *Platt Scaling* (Platt et al. 1999) in Section 4. Minderer et al. (2021) report that modern vision models are well calibrated. Desai and Durrett (2020) study the calibration of pre-trained transformer models, such as *RoBERTa* (Liu et al. 2019). They find that transformer models are generally well calibrated in-domain. Kong et al. (2020) tackle the challenge of getting calibrated out-of-domain scores from pre-trained transformer models using synthetic data.

3 Existing Approaches on Quantification Methods

In the following, we will assume that we are presented with an unlabelled set of text samples Q and tasked to find the

fraction p of samples that could be considered hateful. We further assume that we have access to a real-valued classification function f and a decision threshold τ such that $f(x_i) \geq \tau$ means that the classifier considers the sample x_i to be hateful.

In this work, we consider the case where we have no control over how f was created. In this setting, we propose to partition Q into the sets Q_{calib} and Q_{eval} . We then collect labels for the samples in Q_{calib} which will be used by the different quantification algorithms described below to compute the prevalence estimate p on the remaining samples Q_{eval} .

We will notate $N_{calib} = |Q_{calib}|$ and $N_{eval} = |Q_{eval}|$ for the sizes of the two sets of samples. Furthermore, $s_i = f(x_i)$ is the real-valued output of f for a given sample x_i and $\hat{y}_i = \mathbb{I}[s_i \geq \tau]$ is the predicted label, where $\mathbb{I}$ denotes the indicator function (we regard $\hat{y}_i = 1$ as the classifier stating that the sample contains hate-speech).

We now introduce quantification approaches from existing literature that we adapted to our setting.

Classify and Count (CC). The simplest quantification method is called *classify and count* (CC) (Forman 2008) and consists in averaging the predictions of the given classifier: $p_{CC} = \frac{1}{N_{eval}} \sum_{i=1}^{N_{eval}} \hat{y}_i$.

Adjusted Classify and Count (ACC). As explained in Section 1, the naive CC estimate, p_{CC} , is biased by both the true positive rate (TPR) and false positive rate (FPR) of our classifier f at the given threshold τ . Since we know that $p_{CC} = p(TPR - FPR) + FPR$, we can solve for p . Since, in general, the true values of TPR and FPR are unknown, we compute their point estimates on Q_{calib} : $T\hat{P}R = \frac{\sum_{j=1}^{N_{calib}} \mathbb{I}[\hat{y}_j=1 \wedge y_j=1]}{\sum_{j=1}^{N_{calib}} \mathbb{I}[\hat{y}_j=1]}$ and $F\hat{P}R = \frac{\sum_{j=1}^{N_{calib}} \mathbb{I}[\hat{y}_j=1 \wedge y_j=0]}{\sum_{j=1}^{N_{calib}} \mathbb{I}[\hat{y}_j=0]}$. We then define the *adjusted classify and count* (ACC) (Forman 2008) estimate as: $p_{ACC} = \frac{p_{CC} - F\hat{P}R}{T\hat{P}R - F\hat{P}R}$.

Probabilistic Classify and Count (PCC). For probabilistic classifiers, the score s_i corresponds to the predicted conditional probability of the positive label: $s_i = p(y_i = 1|x_i)$. We can, therefore, average the classifier scores to get the expected fraction of hateful samples: $p_{PCC} = \frac{1}{N_{eval}} \sum_{i=1}^{N_{eval}} s_i$. This method is called *probabilistic classify and count* (PCC) (Forman 2008; Bella et al. 2010) and can only be applied to probabilistic classifiers.

Probabilistic Adjusted Classify and Count (PACC). Analogous to CC, we can define an adjusted version of PCC called *probabilistic adjusted classify and count* (PACC) (Bella et al. 2010). In this case, we rely on probabilistic estimates of TPR and FPR : $T\hat{P}R = \frac{\sum_{j=1}^{N_{calib}} s_j \mathbb{I}[y_j=1]}{\sum_{j=1}^{N_{calib}} \mathbb{I}[y_j=1]}$ and $F\hat{P}R = \frac{\sum_{j=1}^{N_{calib}} s_j \mathbb{I}[y_j=0]}{\sum_{j=1}^{N_{calib}} \mathbb{I}[y_j=0]}$. The resulting estimate of p is: $p_{PACC} = \frac{p_{PCC} - F\hat{P}R}{T\hat{P}R - F\hat{P}R}$.

4 Our Novel Quantification Methods

Next, we introduce our two novel approaches. The two approaches differ in that the first uses calibration to adjust the

Method	Uses τ	Uses Q_{calib}	Assumes probabilistic ratings
CC	x		
ACC	x	x	
PCC			x
PACC		x	x
CPCC		x	
BCC	x	x	

Table 1: Overview of the quantification methods described in Sections 3 and 4.

classifier outputs, and the second extends the *adjusted count* approach by including the uncertainty over TPR and FPR, instead of relying on point estimates.

Calibrated Probabilistic Classify and Count (CPCC).

One of the main points of focus of this work is quantification using a classification function f in cases where the set Q deviates substantially from the training data used to create f . In this setting, a probabilistic classifier f is often improperly calibrated for samples in Q (Guo et al. 2017; Kong et al. 2020). Therefore, we use *Platt-Scaling* (Platt et al. 1999) to re-calibrate f for Q^2 . This means training a logistic regression model on $(f(x_j), y_j)_{j=1}^{N_{calib}}$ and using the probabilistic predictions of this logistic regression model for samples in Q_{eval} as the new scores $\tilde{s}_i$. The resulting prevalence estimate is then: $p_{CPCC} = \frac{1}{N_{eval}} \sum_{i=1}^{N_{eval}} \tilde{s}_i$.

Bayesian Classify and Count (BCC). Recently, von Däniken et al. (2022) introduced a Bayesian model which can be re-purposed for quantification. It was originally developed in the context of automated evaluation of text generation systems. The main idea is to estimate the prior probability distributions of p , TPR , and FPR from Q_{calib} , which can be combined with the predicted labels $\hat{y}_i$ on Q_{eval} . The prevalence estimate p_{BCC} is then the expected value of the resulting posterior distribution.

We summarize the quantification methods described in this section and in Section 3 in Table 1. CPCC and BCC were specifically developed to take advantage of our assumed setting where we have access to a labelled in-domain set of samples Q_{calib} . By design, CC and PCC cannot incorporate the additional information. In contrast, the *adjusted count* methods can easily be adapted to our setting.

5 Evaluation Framework

In this section, we describe the framework for evaluating the proposed quantification methods. We will rely on a range of existing datasets to create various classifiers f^3 . We will also use the test sets of these collections as the quantification

²We compare to *Isotonic Regression* (Zadrozny and Elkan 2002) and *Histogram Binning* (Zadrozny and Elkan 2001) in Appendix A.

³While we train a range of classifiers here, for our evaluation we treat them as if we took them off-the-shelf.

targets Q . We give the details of the datasets, classifiers, and evaluation measures selected for our study below.

5.1 Datasets

We will now give a brief overview of the different datasets used in this study. A summary of their properties is given in Table 2. We relied on hatespeechdata.com maintained by Vidgen and Derczynski (2020) for curating this collection. Our main inclusion criterion was that a dataset should be relatively easily available for download and that its annotations can be converted into a binary label. We aimed to maintain the canonical splits into train C_{train} , development C_{dev} , and test sets C_{test} wherever possible. Datasets that do not have a canonical separation between C_{train} and C_{test} will be used as C_{test} only. If a dataset does not have a development set C_{dev} , we use 10% of the training data as C_{dev} . Our experimental setting comprises a highly diverse set of 11 datasets with different domains, annotation schemes, and prevalence of offensive language.

Davidson. Davidson et al. (2017) collected tweets from users who had used words and phrases from a hate speech lexicon. The tweets were annotated by crowd-workers as *hate*, *offensive*, or *neither*. The main distinction between *hate* and merely *offensive* tweets is that hateful tweets express explicit hatred towards a target group. For our experiments, we consider the case where *hate* and *offensive* together constitute the positive class, which we call *Davidson Hate & Offensive*. We also consider the case where only the *hate* class constitutes the positive class and call this *Davidson Hate Only*.

Dynamically Generated Hate (DGH). Vidgen et al. (2021) set out to create a dataset to train robust classifiers. Starting from a classifier trained on multiple existing hate speech datasets, they proceed in multiple rounds. In each round, they ask annotators to produce texts that trigger misclassifications from the model. A new model is then re-trained based on this data. They annotated both whether a text is hateful as well as the type and target of hate. In this work, we rely on the top-level binary hate label.

ETHOS. The *ETHOS* (Mollas et al. 2022) dataset comes in both a binary and a multi-label version. It consists of comments from *YouTube* and *Reddit*. The main goal was to create a balanced multi-label dataset covering aspects such as race, gender, and religion, as well as whether a comment is directed or generalized, or incites violence. We consider positive cases those where strictly more than half of the annotators consider a comment as hateful. Due to its limited size, we use it only as a test set.

Ex Machina. Wulczyn, Thain, and Dixon (2017) study personal attacks in discussions on *Wikipedia*. They had crowd workers annotate random comments, as well as comments sampled specifically from users that were banned for personal attacks. Each comment was annotated by 10 workers. In our experiments, we consider positive cases those where more than half of the annotators consider a comment a personal attack.

HASOC 2019. The *HASOC 2019* (Mandl et al. 2019) shared task contained three sub-tasks. The first sub-task consisted of distinguishing hateful and offensive content from non-

offensive posts. The other sub-tasks involved classifying the type of hate or offense and its target. The task organizers provided datasets in English, German, and Hindi. In this work, we rely on the binary labels and the English dataset.

HatEval 2019. The *HatEval 2019* (Basile et al. 2019) shared task focused on hate that targets women and immigrants. The main sub-task involved classifying Twitter posts into hateful and not hateful. The other sub-tasks were predicting the specificity of the target and aggressiveness of the tweet. The organizers provided data in English and Spanish. We focus on the binary annotations and English data.

Jigsaw. The *Jigsaw Toxic Comment Classification Challenge* (cjadams et al. 2017) is a popular dataset on *Kaggle*⁴, an online platform that hosts machine learning challenges. The dataset contains comments from Wikipedia discussions annotated with multiple binary labels: toxic, severely toxic, obscene, threat, insult, and identity hate. In this work, we assign a positive label if any of these labels are present.

OLID. The *Offensive Language Identification Dataset (OLID)* (Zampieri et al. 2019a) contains layered annotations whether a tweet contains offensive language, whether the offense is targeted, and whether the target is an individual or a group. The dataset has been used to run the *OffensEval 2019* (Zampieri et al. 2019b) shared task. In this work, we rely on the top-level binary offensiveness labels.

SOLID. The *Semi-Supervised Offensive Language Identification Dataset (SOLID)* (Rosenthal et al. 2021) is a semi-supervised extension of *OLID*. The annotation scheme follows *OLID*, and they provide a large corpus of English tweets with automatic annotations. In addition, they include a larger manually annotated test set. The *SOLID* data was used for the *OffensEval 2020* (Zampieri et al. 2020) shared task. In this work, we will only use the manually annotated test set.

SWAD. The *Swear Word Abusiveness Dataset (SWAD)* (Pamungkas, Basile, and Patti 2020) is also based on *OLID*. The authors provide new annotations indicating whether the use of a specific swear word makes a post abusive or not. We use these binary labels as an additional test set.

WASSA. Grimminger and Klinger (2021) created a dataset for both stance detection and hatefulness and offensiveness. It consists of tweets about the candidates of the 2020 US elections. We focus on the binary hate / offensiveness label.

5.2 Classifiers

Next, we describe the classifiers we use in our experiments. We train three types of classifiers on each training set individually and collect their predictions for all test sets. The aim is to use a diverse set of classifiers in terms of architectures, models, and performance to showcase the generality of our quantification approach.

Electra. *Electra* (Clark et al. 2020) models have the same architecture as *BERT* (Devlin et al. 2019), but follow a different pre-training procedure. During masked language modeling (MLM) pre-training, there are two models: a generator G and a discriminator D . G is trained to predict the

⁴<https://kaggle.com>

Dataset	Size Train	Size Dev	Size Test	Prev. Train	Prev. Dev	Prev. Test
Davidson Hate & Offensive	22304	-	2479	83.2%	-	83.2%
Davidson Hate Only	22304	-	2479	5.8%	-	5.8%
DGH	32924	4100	4120	53.9%	52.9%	55.0%
ETHOS	-	-	998	-	-	43.4%
Ex Machina	69526	23160	23178	13.4%	13.8%	13.2%
HASOC 2019	5852	-	1153	38.6%	-	25.0%
HatEval 2019	9000	1000	3000	42.0%	42.7%	42.0%
Jigsaw	159571	-	63978	10.2%	-	9.8%
OLID	13240	-	860	33.2%	-	27.9%
SOLID	-	-	5993	-	-	50.1%
SWAD	-	-	2578	-	-	32.7%
WASSA	2400	-	600	12.2%	-	9.8%

Table 2: Overview of the datasets used for our experiments. We report the prevalence of the positive class for every dataset as described in Section 5.1.

masked tokens. The masked tokens are then replaced by the top prediction of G , and D is trained to predict which tokens were generated by G and which ones are original.

For our experiments, we use a discriminator model checkpoint⁵ from *Hugging Face*. We train the model for up to 50 epochs with batch size 16, stopping early if the loss on C_{dev} does not improve for 5 epochs. The model weights are updated using *AdamW* (Loshchilov and Hutter 2019) with a learning rate of $5e-5$ and weight decay of 0.01. Texts that are longer than the maximum of 512 tokens are truncated to that length.

Twitter-RoBERTa. Since many datasets consist of tweets, we include classifiers pre-trained on Twitter data. Specifically, we fine-tune a *RoBERTa* (Liu et al. 2019) model that was pre-trained by Barbieri et al. (2020) on 58M tweets⁶. Our training procedure is identical to the *Electra* models.

TF-IDF SVM. We train *Support Vector Machine* (SVM) (Cortes and Vapnik 1995) models on *TF-IDF* (Manning, Raghavan, and Schütze 2008) features. We use the *TfidfVectorizer* and *LinearSVC* implementations provided by *scikit-learn* (Pedregosa et al. 2011). During training, we perform a grid search over three hyper-parameters and select the model that has the highest F1 score on C_{dev} . The hyperparameter ranges are: maximum n-gram length (2, 3, 5, or 7), whether to use binary term frequencies or log-scale them, and the regularization strength of the SVM loss (0.001, 0.01, 0.1, 1, 10, 100, or 1000).

Perspective API. Further, we collect predictions from an online service⁷. Since the organisations developing this service are also the providers of the *Jigsaw* dataset, its predictions are in line with the *Jigsaw* multi-label annotations. It will predict a score between 0 and 1 for each of the *Jigsaw* labels. We will use the maximum score over all labels and a decision threshold of $\frac{1}{2}$.

⁵<https://huggingface.co/google/electra-base-discriminator>

⁶<https://huggingface.co/cardiffnlp/twitter-roberta-base>

⁷<https://www.perspectiveapi.com/>

N_{calib}	10	20	30	40	50	60	70+
Random	0.13	0.08	0.03	0.02	0.01	0.01	0.00
Quantile	0.10	0.02	0.00	0.00	0.00	0.00	0.00

Table 3: Fraction of cases where our selection strategies do not produce any positive samples depending on the number N_{calib} of samples selected.

5.3 Selecting Samples to Annotate

As described in Section 3, we select a certain number of samples of the unlabelled quantification set Q to be labelled and then compute the prevalence estimate on the remaining samples. We will explore two methods of selecting samples for annotation to partition Q into Q_{calib} and Q_{eval} .

The first sample selection method we will explore is to select N_{calib} random samples to annotate from Q_{calib} . We will call this method *random*. The other method is based on the classifier scores $s_i = f(x_i)$. We first sort the samples by their score s_i . We then split the sorted samples into $N_q = 10$ consecutive equal length segments. We then select $\frac{N_{calib}}{N_q}$ random samples from each segment to form Q_{calib} . We will call this method *quantile*, since each of the N_q segments correspond to a quantile of the score distribution.

Ideally, our selection method will produce a Q_{calib} that is representative of the whole Q , which is a pre-requisite for the methods in Sections 3 and 4 to work. A particular failure mode of our selection strategies is when they fail to select any positive samples at all. In this case, we cannot properly estimate the TPR and FPR of f , and calibration using *Platt Scaling* is also impossible.

In Table 3, we show the fraction of cases in which neither selection method produces any positive samples for various values of N_{calib} . For this, we applied each sample selection strategy to each of our 12 test sets 10 times with different random seeds. In general, we observe that the *quantile* strategy produces fewer failures and both strategies succeed when $N_{calib} \geq 70$. Based on this, we will use $N_{calib} = 100$ for our experiments, unless otherwise stated.

One other factor that influences the success of our selec-

p	0.01	0.02	0.03	0.05	0.07	0.10
Random	0.26	0.11	0.02	0.03	0.00	0.00
Quantile	0.36	0.16	0.09	0.02	0.00	0.00

Table 4: Fraction of cases where our selection strategies do not produce any positive samples depending on the prevalence p in the target set Q .

tion methods is the true prevalence p . In general, the smaller p is, the harder it is to select positives. In Table 2 we can see that the smallest p for any of our test sets is 9.8%. To see how often our selection methods fail for a small p , we sub-sample the positive class of our test sets to a number of fixed prevalences. In cases where sub-sampling would lead to fewer than 15 positives, we remove the test set from consideration for that given prevalence. Table 4 shows the fraction of cases where either selection method does not produce any positive samples for different values of p . In this case, we fixed $N_{calib} = 100$. We can see that as long as the true prevalence $p > 3\%$, both selection methods produce only few or no failures.

In what follows, we will generally operate in the regime where there are very few or no failures during the sample selection process and any sample selection failures that do occur will be excluded from the analysis.

5.4 Performance Measures

To measure the performance of a given quantification method given an unlabelled set Q and classifier f (with threshold τ), we first apply a sample selection method from Section 5.3, resulting in $Q_{eval} \cup Q_{calib} = Q$. Let p_x be the estimated prevalence produced by a method described in Section 3 or 4 and p be the true prevalence in Q_{eval} .

Many different measures for the quality of a quantification approach exist (González et al. 2017). In this work, we will use the *symmetric absolute percentage error (SAPE)* which is defined as $\frac{|p_x - p|}{p_x + p}$. The SAPE is bounded and ranges from 0 to 1 with lower values indicating better quantification performance. In particular, it captures the intuition that the severity of some constant absolute error depends on the true underlying prevalence, i.e., an estimate that is 0.01 off is less severe when the true prevalence is 0.50 than if the true prevalence were 0.02.

6 Experiments

In this section, we show the results of our experiments. We fine-tune the three trainable models on each of the datasets where a train set C_{train} is available (see Table 2) and generate the outputs for each of the test sets. Overall, we note that the sample-based performances are highly diverse and the ROC-AUC scores range from 0.486 to 0.989 depending on the classifier, train and test set combination (see Appendix B). Overall, we train 3 classifiers on 9 training sets and add 1 online service for a total of 28 classifiers f . For each model, we then apply each of the quantification methods and the test splits serve as the unlabelled data Q . In the following, we will show the main results comparing quan-

	Random			Quantile		
	μ	Med.	95	μ	Med.	95
CC	0.368	0.264	1.000	0.368	0.264	1.000
ACC	0.352	0.230	1.000	0.240	0.125	1.000
PCC	0.299	0.220	0.827	0.299	0.219	0.827
PACC	0.306	0.167	1.000	0.151	0.077	0.635
CPCC	0.073	0.049	0.211	0.085	0.059	0.275
BCC	0.067	0.046	0.209	0.082	0.059	0.259

Table 5: SAPE of different quantification methods. We report the mean (μ), median (Med.), and 95th percentile (95) of errors for all quantification methods and sample selection methods.

tification approaches and sample selection strategies and explore other factors.

6.1 Comparing Quantification Methods

Here, we compare the performance of the quantification methods described in Section 4. We also include methods from Section 3 as baselines. We applied each quantification approach to each pair of classifier and test set, using a set of $N_{calib} = 100$ in-domain calibration samples. We ran each experiment 10 times with different random seeds, yielding 28 classifiers $\times$ 12 test sets $\times$ 10 results for each quantification approach (i.e., 3360 results). In Table 5, we show the mean, median, and 95th percentile SAPE for each quantification method and sample selection strategy.

The results show that our novel approaches *CPCC* and *BCC* markedly outperform the existing quantification approaches. They achieve a median SAPE of 0.049 and 0.046, respectively, for *random* sample selection, and both achieve a median SAPE of 0.059 for *quantile* selection. Out of the existing approaches, *PACC* in combination with *quantile* sampling performs best, achieving a median SAPE of 0.077. However, when using the *random* sampling approach, the error rate of *PACC* rises substantially to a SAPE of 0.167. The naive *classify and count* approach only achieves high median SAPE scores of 0.264. Furthermore, the variance is much lower for our novel approaches. In fact, *CPCC* and *BCC* have a 95th percentile SAPE of 0.211 and 0.209, respectively, for *random* sample selection. In contrast, the 95th percentile SAPE of the other approaches ranges from 0.635 to 1.0.

Thus, our novel quantification approaches achieve superior performance scores and are also more robust in regard to domains and underlying classifiers.

6.2 Out-of-Domain Calibration

In the above experiment, we used $N_{calib} = 100$ in-domain calibration samples. Here, we showcase the necessity for these by repeating the same type of experiment but using an entire out-of-domain dataset for calibration. This corresponds to the setting where we rely on a different pre-existing set of labelled samples (for example from a shared task) instead of selecting a number of samples from Q to be labelled. We simulate this scenario by pairing all test splits in Table 2 with each other to build (Q_{calib}, Q_{eval}) pairs (such

	Out-of-Domain		
	μ	Med.	95
CC	0.368	0.262	1.000
ACC	0.582	0.520	1.000
PCC	0.299	0.221	0.829
PACC	0.575	0.520	1.000
CPCC	0.357	0.309	0.815
BCC	0.394	0.365	0.792

Table 6: *SAPE* of different quantification methods when using *out-of-domain* data as Q_{calib} . We report the mean (μ), median (Med.), and 95th percentile (95) of errors for all quantification methods and sample selection methods.

	Random			Quantile		
	μ	Med.	95	μ	Med.	95
PACC-10	0.424	0.294	1.000	0.249	0.163	1.000
PACC-20	0.411	0.289	1.000	0.230	0.142	1.000
PACC-40	0.366	0.246	1.000	0.207	0.117	0.892
PACC-80	0.311	0.173	1.000	0.158	0.081	0.654
PACC-160	0.272	0.136	1.000	0.131	0.065	0.519
CPCC-10	0.175	0.126	0.505	0.180	0.132	0.511
CPCC-20	0.140	0.102	0.385	0.151	0.111	0.432
CPCC-40	0.120	0.079	0.399	0.132	0.088	0.400
CPCC-80	0.080	0.056	0.241	0.094	0.063	0.279
CPCC-160	0.051	0.034	0.158	0.073	0.049	0.224
BCC-10	0.157	0.120	0.441	0.174	0.132	0.486
BCC-20	0.131	0.101	0.358	0.141	0.108	0.396
BCC-40	0.103	0.075	0.342	0.122	0.089	0.368
BCC-80	0.072	0.053	0.216	0.089	0.062	0.264
BCC-160	0.048	0.032	0.160	0.071	0.049	0.219

Table 7: *SAPE* of different quantification methods depending on the number of samples selected for calibration. Sample sizes are indicated as a suffix to the name of the quantification method. We report the mean (μ), median (Med.), and 95th percentile (95) of errors for all quantification methods and sample selection methods.

that $Q_{calib} \neq Q_{test}$, e.g., calibrate on SWAD and test on OLID). We use all 28 classifiers and repeat each experiment 10 times with different random seeds.

The results show that each of the quantification approaches yield relatively high median errors. In fact, none of the approaches that rely on Q_{calib} outperform the ones that do not (CC and PCC). This is to be expected, as an out-of-domain dataset can yield mismatched TPR and FPR estimates. Similarly, CPCC is influenced by the inconsistent prevalence between datasets. This explicitly showcases the need to use in-domain samples for calibration.

6.3 Reducing the Number of Samples to Select

Since using out-of-domain data is inadequate, we now investigate to which degree the number of samples N_{calib} to annotate can be reduced. For this, we repeat the experiments from Table 5 with different values of $N_{calib} \in \{10, 20, 40, 80, 160\}$ and compute the resulting *SAPE* for the best-performing methods PACC, CPCC, and BCC. We

have already mentioned that reducing N_{calib} can lead to an increase in cases where our sample selection method fails to find positive samples (see Section 5.3). In these cases, we dropped the experiment configuration from the analysis, meaning that those cases do not contribute to the error statistics.

Table 7 shows the results of the sample-reduction experiments. For CPCC, the median *SAPE* ranges from 0.034 for 160 samples to 0.126 for 10 samples. For BCC the median *SAPE* ranges from 0.032 for 160 samples to 0.120 for 10 samples. Similar to the main results, the *random* selection method performs better than the *quantile* selection. The range is larger for PACC (the median *SAPE* lies between 0.065 and 0.163), and similar to the main results, *quantile* outperforms the *random* selection method.

We note that both CPCC and BCC outperform CC, ACC, and PCC (see Table 5) using only 10 samples. For the random selection method, CPCC-40 and BCC-40 are on par with PACC-80 (0.079, 0.075, and 0.081, respectively). Considering only the 95th percentile, CPCC-10 and BCC-10 outperform PACC-160 (0.505, 0.441, and 0.519 respectively).

Overall, all methods perform better for larger N_{calib} . The performance of CPCC and BCC degrades relatively gracefully. The sample selection method has a large influence on PACC, whereas CPCC and BCC seem relatively stable with *random* selection performing slightly better.

6.4 Low Prevalence

To study how our methods perform when decreasing the true prevalence of hateful content, we proceed in the same way as in Section 5.3. For each target prevalence, we sub-sample the positive class to that level, making sure there are at least 15 positives. If there are fewer positives when sub-sampling, we ignore the configuration. Similarly, we ignore cases where our sample selection strategy (using $N_{calib} = 100$) does not produce any positives in D_{calib} . We run each experiment 10 times with different random seeds.

We show the *SAPE* for CC, PACC, CPCC, and BCC in Tables 8 and 9. For all methods the *SAPE* increases as the prevalence p decreases. This is to be expected, as the denominator in the *SAPE* calculation gets smaller. The median *SAPE* for CPCC with *random* sample selection ranges from 0.092 to 0.206. For BCC the median *SAPE* ranges from 0.090 to 0.178. Similar to previous results, *quantile* sample selection performs worse for CPCC and BCC. For PACC, *quantile* selection leads to lower median *SAPE*, ranging from 0.165 to 0.361. We note that in the low prevalence setting, the performance difference between PACC and our novel methods is even more pronounced. This is also apparent when considering the 95th percentile *SAPE* which is 1.0 for all prevalences for PACC.

7 Discussion & Conclusion

Our setting is motivated by a real-world scenario in which black-box classifiers are used without the possibility to fine-tune. Furthermore, our scenario also assumes that there is only a very small set of annotations available making classical domain-transfer extremely difficult.

p	CC			PACC			CPCC			BCC		
	μ	Med.	95	μ	Med.	95	μ	Med.	95	μ	Med.	95
0.020	0.716	0.780	1.000	0.706	0.811	1.000	0.215	0.206	0.527	0.196	0.178	0.523
0.030	0.648	0.724	1.000	0.629	0.677	1.000	0.189	0.157	0.529	0.169	0.155	0.358
0.050	0.578	0.630	1.000	0.542	0.494	1.000	0.179	0.136	0.476	0.158	0.137	0.389
0.070	0.499	0.524	1.000	0.490	0.398	1.000	0.128	0.107	0.298	0.115	0.093	0.275
0.100	0.464	0.481	1.000	0.411	0.302	1.000	0.109	0.092	0.274	0.106	0.090	0.243

Table 8: *SAPE* of different quantification methods with *random* sample selection depending on the prevalence p . We report the mean (μ), median (Med.), and 95th percentile (95) of errors for all quantification methods and sample selection methods.

p	CC			PACC			CPCC			BCC		
	μ	Med.	95	μ	Med.	95	μ	Med.	95	μ	Med.	95
0.020	0.706	0.772	1.000	0.458	0.361	1.000	0.219	0.210	0.473	0.195	0.178	0.473
0.030	0.645	0.715	1.000	0.440	0.333	1.000	0.237	0.205	0.551	0.201	0.176	0.480
0.050	0.575	0.624	1.000	0.344	0.240	1.000	0.201	0.144	0.635	0.180	0.151	0.455
0.070	0.501	0.525	1.000	0.299	0.187	1.000	0.170	0.129	0.486	0.156	0.123	0.418
0.100	0.465	0.483	1.000	0.266	0.165	1.000	0.153	0.117	0.440	0.141	0.109	0.391

Table 9: *SAPE* of different quantification methods with *quantile* sample selection depending on the prevalence p . We report the mean (μ), median (Med.), and 95th percentile (95) of errors for all quantification methods and sample selection methods.

In Sections 5.1 and 5.2, we described how we train various classification models for our experiments following the standard procedure of using train (C_{train}) and development data (C_{dev}). In the usual classification setting, one would then evaluate f on the test data C_{test} . In this work, we use C_{test} as the quantification target data Q . This leads to the following issue: our method relies on splitting Q into Q_{calib} and Q_{eval} and, crucially, we assume access to the labels of Q_{calib} in order to calibrate the predictions of f . In the traditional setting, this leads to an unfair advantage. Of course, in this work, this advantage is entirely intentional and we adjusted previous and proposed quantification methods accordingly (see Table 5). We could, instead, have opted to try using C_{dev} as our calibration set Q_{calib} . One issue is that in the real world there might not be any annotated data a-priori for a domain of interest. Further, we have shown in Section 6.2 that using out-of-domain data for calibration yields poor results. We can see in Table 2 that the prevalence of hateful content can differ even between different splits of the same dataset, e.g. in the cases of *HASOC 2019* and *OLID*.

Our experiments show that it is possible to get good quantification estimates with only 100 annotated in-domain samples. The results are robust over a variety of classifiers and datasets, in particular when considering that our datasets had annotations for related but notably different tasks.

It should, therefore, in principle be possible to study the prevalence of hateful content in a new dataset using existing off-the-shelf classifiers. The main motivation to use classifiers for this purpose in the first place is to eschew the laborious annotation process. Our method yields markedly improved results while keeping the annotation effort low.

We note that inaccurate quantification approaches can pose significant risks, as they can lead to wildly inaccurate prevalence estimates. For example, in Table 8 we see that the

naive *CC* approach has a median *SAPE* of 0.481 for the moderately low prevalence $p = 0.1$. This means that the estimate under *CC* could be as low as 0.035 or up to 0.285. Conclusions based on such poor estimates can be highly misleading. This can lead to misdirected policies and interventions or a misallocation of resources.

An important future extension of this work is considering how we can properly compare the prevalence of hateful content between two collections. In this case, we would want to gain access to a statistical test that can tell us whether one collection contains significantly more such content than another. This means that we would have to go beyond point estimates of p and consider confidence intervals. Interestingly, *BCC* in principle produces a full posterior distribution that can be used for this (von Däniken et al. 2022).

While our methods could in principle be applied to other binary quantification tasks, this work is limited to hateful, abusive, and toxic language in English. We have relied entirely on existing datasets that are publicly available. Since the main focus of this work were quantification methods, no one had to assess the content of the datasets in detail. Therefore, no one had to come into contact with abusive material or potentially personally identifiable information.

In the regime of very low prevalences our method runs into the particular issue that selecting samples to build D_{calib} can fail. In this work, we focused on evaluating the quantification methods. Better sample selection strategies could be developed based on active learning (Kumar and Gupta 2020) and rare class discovery methods (He and Carbonell 2008).

Acknowledgments

This work was supported by the CHIST-ERA HAMi-SoN project grant CHIST-ERA-21-OSNEM002, by SNF

20CH21 209672 and AEI PCI2022-135026-2.

References

- Ali, R.; Farooq, U.; Arshad, U.; Shahzad, W.; and Beg, M. O. 2022. Hate speech detection on Twitter using transfer learning. *Computer Speech & Language*, 74: 101365.
- Aragón, M. E.; Carmona, M. A. A.; Montes-y Gómez, M.; Escalante, H. J.; Pineda, L. V.; and Moctezuma, D. 2019. Overview of MEX-A3T at IberLEF 2019: Authorship and Aggressiveness Analysis in Mexican Spanish Tweets. In *IberLEF@ SEPLN*, 478–494.
- Aragón, M. E.; Jarquín-Vásquez, H. J.; Montes-y Gómez, M.; Escalante, H. J.; Pineda, L. V.; Gómez-Adorno, H.; Posadas-Durán, J. P.; and Bel-Enguix, G. 2020. Overview of MEX-A3T at IberLEF 2020: Fake News and Aggressiveness Analysis in Mexican Spanish. In *IberLEF@ SEPLN*, 222–235.
- Barbieri, F.; Camacho-Collados, J.; Espinosa Anke, L.; and Neves, L. 2020. TweetEval: Unified Benchmark and Comparative Evaluation for Tweet Classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 1644–1650. Online: Association for Computational Linguistics.
- Basile, V.; Bosco, C.; Fersini, E.; Nozza, D.; Patti, V.; Rangel Pardo, F. M.; Rosso, P.; and Sanguinetti, M. 2019. SemEval-2019 Task 5: Multilingual Detection of Hate Speech Against Immigrants and Women in Twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, 54–63. Minneapolis, Minnesota, USA: Association for Computational Linguistics.
- Bella, A.; Ferri, C.; Hernandez-Orallo, J.; and Ramirez-Quintana, M. J. 2010. Quantification via Probability Estimators. In *Proceedings of the 2010 IEEE International Conference on Data Mining, ICDM '10*, 737–742. USA: IEEE Computer Society. ISBN 9780769542560.
- Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; Herbert-Voss, A.; Krueger, G.; Henighan, T.; Child, R.; Ramesh, A.; Ziegler, D.; Wu, J.; Winter, C.; Hesse, C.; Chen, M.; Sigler, E.; Litwin, M.; Gray, S.; Chess, B.; Clark, J.; Berner, C.; McCandlish, S.; Radford, A.; Sutskever, I.; and Amodei, D. 2020. Language Models are Few-Shot Learners. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 1877–1901. Curran Associates, Inc.
- cjadams, J.; Sorensen; Elliott, J.; Dixon, L.; McDonald, M.; nithum; and Cukierski, W. 2017. Toxic Comment Classification Challenge.
- Clark, K.; Luong, M.; Le, Q. V.; and Manning, C. D. 2020. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.
- Cortes, C.; and Vapnik, V. N. 1995. Support-Vector Networks. *Machine Learning*, 20: 273–297.
- Davidson, T.; Warmley, D.; Macy, M.; and Weber, I. 2017. Automated Hate Speech Detection and the Problem of Offensive Language. *Proceedings of the International AAAI Conference on Web and Social Media*, 11(1): 512–515.
- Deriu, J. M.; Weilenmann, M.; Von Gruenigen, D.; and Cieliebak, M. 2017. Potential and Limitations of Cross-Domain Sentiment Classification. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, 17–24. Valencia, Spain: Association for Computational Linguistics.
- Desai, S.; and Durrett, G. 2020. Calibration of Pre-trained Transformers. In Webber, B.; Cohn, T.; He, Y.; and Liu, Y., eds., *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 295–302. Online: Association for Computational Linguistics.
- Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics.
- Esuli, A.; Moreo, A.; Sebastiani, F.; and Sperduti, G. 2022. A Concise Overview of LeQua@CLEF 2022: Learning to Quantify. In Barrón-Cedeño, A.; Da San Martino, G.; Degli Esposti, M.; Sebastiani, F.; Macdonald, C.; Pasi, G.; Hanbury, A.; Potthast, M.; Faggioli, G.; and Ferro, N., eds., *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, 362–381. Cham: Springer International Publishing. ISBN 978-3-031-13643-6.
- Esuli, A.; and Sebastiani, F. 2015. Optimizing Text Quantifiers for Multivariate Loss Functions. *ACM Trans. Knowl. Discov. Data*, 9(4).
- FORCE11. 2020. The FAIR Data principles. <https://force11.org/info/the-fair-data-principles/>.
- Forman, G. 2005. Counting Positives Accurately Despite Inaccurate Classification. In Gama, J.; Camacho, R.; Brazdil, P. B.; Jorge, A. M.; and Torgo, L., eds., *Machine Learning: ECML 2005*, 564–575. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN 978-3-540-31692-3.
- Forman, G. 2008. Quantifying counts and costs via classification. *Data Mining and Knowledge Discovery*.
- Geburu, T.; Morgenstern, J.; Vecchione, B.; Vaughan, J. W.; Wallach, H.; Iii, H. D.; and Crawford, K. 2021. Datasheets for datasets. *Communications of the ACM*, 64(12): 86–92.
- González, P.; Castaño, A.; Chawla, N. V.; and Coz, J. J. D. 2017. A Review on Quantification Learning. *ACM Comput. Surv.*, 50(5).
- Grimminger, L.; and Klinger, R. 2021. Hate Towards the Political Opponent: A Twitter Corpus Study of the 2020 US Elections on the Basis of Offensive Speech and Stance Detection. In *Proceedings of the Eleventh Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 171–180. Online: Association for Computational Linguistics.

- Guo, C.; Pleiss, G.; Sun, Y.; and Weinberger, K. Q. 2017. On Calibration of Modern Neural Networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, 1321–1330. JMLR.org.
- Hajibabae, P.; Malekzadeh, M.; Ahmadi, M.; Heidari, M.; Esmaeilzadeh, A.; Abdolazimi, R.; and Jones, J. H. J. 2022. Offensive Language Detection on Social Media Based on Text Classification. In *2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*, 0092–0098.
- He, J.; and Carbonell, J. 2008. Rare class discovery based on active learning. 7P. 10th International Symposium on Artificial Intelligence and Mathematics, ISAIM 2008 ; Conference date: 02-01-2008 Through 04-01-2008.
- Joachims, T. 2005. A Support Vector Method for Multi-variate Performance Measures. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05, 377–384. New York, NY, USA: Association for Computing Machinery. ISBN 1595931805.
- Kong, L.; Jiang, H.; Zhuang, Y.; Lyu, J.; Zhao, T.; and Zhang, C. 2020. Calibrated Language Model Fine-Tuning for In- and Out-of-Distribution Data. In Webber, B.; Cohn, T.; He, Y.; and Liu, Y., eds., *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1326–1340. Online: Association for Computational Linguistics.
- Kumar, P.; and Gupta, A. 2020. Active Learning Query Strategies for Classification, Regression, and Clustering: A Survey. *Journal of Computer Science and Technology*, 35(4): 913–945.
- Liu, J.; Kong, D.; Huang, L.; Mao, D.; and Xue, H. 2022. Multiple Instance Learning for Offensive Language Detection. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, 7387–7396. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics.
- Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv:1907.11692.
- Loshchilov, I.; and Hutter, F. 2019. Decoupled Weight Decay Regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.
- Mandl, T.; Modha, S.; Majumder, P.; Patel, D.; Dave, M.; Mandlia, C.; and Patel, A. 2019. Overview of the HASOC Track at FIRE 2019: Hate Speech and Offensive Content Identification in Indo-European Languages. In *Proceedings of the 11th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE '19*, 14–17. New York, NY, USA: Association for Computing Machinery. ISBN 9781450377508.
- Manning, C. D.; Raghavan, P.; and Schütze, H. 2008. *Introduction to Information Retrieval*. Cambridge University Press.
- Markov, I.; and Daelemans, W. 2021. Improving Cross-Domain Hate Speech Detection by Reducing the False Positive Rate. In *Proceedings of the Fourth Workshop on NLP for Internet Freedom: Censorship, Disinformation, and Propaganda*, 17–22. Online: Association for Computational Linguistics.
- Minderer, M.; Djolonga, J.; Romijnders, R.; Hubis, F.; Zhai, X.; Houlsby, N.; Tran, D.; and Lucic, M. 2021. Revisiting the Calibration of Modern Neural Networks. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems*, volume 34, 15682–15694. Curran Associates, Inc.
- Mollas, I.; Chrysopoulou, Z.; Karlos, S.; and Tsoumakas, G. 2022. ETHOS: a multi-label hate speech detection dataset. *Complex & Intelligent Systems*, 8(6): 4663–4678.
- Mubarak, H.; Darwish, K.; Magdy, W.; Elsayed, T.; and Al-Khalifa, H. 2020. Overview of OSACT4 Arabic Offensive Language Detection Shared Task. In *4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, 48–52. Marseille, France: European Language Resource Association. ISBN 979-10-95546-51-1.
- Nakov, P.; Ritter, A.; Rosenthal, S.; Sebastiani, F.; and Stoyanov, V. 2016. SemEval-2016 Task 4: Sentiment Analysis in Twitter. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, 1–18. San Diego, California: Association for Computational Linguistics.
- Pamungkas, E. W.; Basile, V.; and Patti, V. 2020. Do You Really Want to Hurt Me? Predicting Abusive Swearing in Social Media. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, 6237–6246. Marseille, France: European Language Resources Association. ISBN 979-10-95546-34-4.
- Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; and Duchesnay, E. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830.
- Platt, J.; et al. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3): 61–74.
- Plaza-del Arco, F. M.; Casavantes, M.; Escalante, H. J.; Martín-Valdivia, M. T.; Montejó-Ráez, A.; Montes, M.; Jarquín-Vásquez, H.; Villaseñor-Pineda, L.; et al. 2021. Overview of MeOffendEs at IberLEF 2021: Offensive language detection in Spanish variants. *Procesamiento del Lenguaje Natural*, 67: 183–194.
- Plaza-del Arco, F. M.; Molina-González, M. D.; Ureña-López, L. A.; and Martín-Valdivia, M.-T. 2022. Integrating implicit and explicit linguistic phenomena via multi-task learning for offensive language detection. *Knowledge-Based Systems*, 258: 109965.
- Rosenthal, S.; Atanasova, P.; Karadzhov, G.; Zampieri, M.; and Nakov, P. 2021. SOLID: A Large-Scale Semi-Supervised Dataset for Offensive Language Identification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 915–928. Online: Association for Computational Linguistics.

- Rosenthal, S.; Farra, N.; and Nakov, P. 2017. SemEval-2017 Task 4: Sentiment Analysis in Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 502–518. Vancouver, Canada: Association for Computational Linguistics.
- Roy, P. K.; Bhawal, S.; and Subalalitha, C. N. 2022. Hate speech and offensive language detection in Dravidian languages using deep ensemble framework. *Computer Speech & Language*, 75: 101386.
- Siegel, A. A.; Nikitin, E.; Barberá, P.; Sterling, J.; Pullen, B.; Bonneau, R.; Nagler, J.; and Tucker, J. A. 2018. Measuring the prevalence of online hate speech, with an application to the 2016 US election.
- Solovev, K.; and Pröllochs, N. 2022. Hate Speech in the Political Discourse on Social Media: Disparities Across Parties, Gender, and Ethnicity. In *Proceedings of the ACM Web Conference 2022*, WWW '22, 3656–3661. New York, NY, USA: Association for Computing Machinery. ISBN 9781450390965.
- Talat, Z.; Thorne, J.; and Bingel, J. 2018. *Bridging the Gaps: Multi Task Learning for Domain Transfer of Hate Speech Detection*, 29–55. Cham: Springer International Publishing. ISBN 978-3-319-78583-7.
- Toraman, C.; Şahinuç, F.; and Yilmaz, E. 2022. Large-Scale Hate Speech Detection with Cross-Domain Transfer. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, 2215–2225. Marseille, France: European Language Resources Association.
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All You Need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, 6000–6010. Red Hook, NY, USA: Curran Associates Inc. ISBN 9781510860964.
- Vidgen, B.; and Derczynski, L. 2020. Directions in abusive language training data, a systematic review: Garbage in, garbage out. *PLOS ONE*, 15(12): e0243300.
- Vidgen, B.; Harris, A.; Nguyen, D.; Tromble, R.; Hale, S.; and Margetts, H. 2019. Challenges and frontiers in abusive content detection. In *Proceedings of the Third Workshop on Abusive Language Online*, 80–93. Florence, Italy: Association for Computational Linguistics.
- Vidgen, B.; Thrush, T.; Waseem, Z.; and Kiela, D. 2021. Learning from the Worst: Dynamically Generated Datasets to Improve Online Hate Detection. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 1667–1682. Online: Association for Computational Linguistics.
- von Däniken, P.; Deriu, J.; Tuggener, D.; and Cieliebak, M. 2022. On the Effectiveness of Automated Metrics for Text Generation Systems. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, 1503–1522. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics.
- Wiegand, M.; Siegel, M.; and Ruppenhofer, J. 2019. Overview of the GermEval 2018 Shared Task on the Identification of Offensive Language. In *Proceedings of GermEval 2018, 14th Conference on Natural Language Processing (KONVENS 2018)*, Proceedings of GermEval 2018, 14th Conference on Natural Language Processing (KONVENS 2018), Vienna, Austria – September 21, 2018, 1–10. Vienna, Austria: Austrian Academy of Sciences. ISBN 978-3-7001-8435-5.
- Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; and Brew, J. 2019. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *CoRR*, abs/1910.03771.
- Wulczyn, E.; Thain, N.; and Dixon, L. 2017. Ex Machina: Personal Attacks Seen at Scale. WWW '17, 1391–1399. Republic and Canton of Geneva, CHE: International World Wide Web Conferences Steering Committee. ISBN 9781450349130.
- Zadrozny, B.; and Elkan, C. 2001. Obtaining calibrated probability estimates from decision trees and naive Bayesian classifiers. In Brodley, C. E.; and Danyluk, A. P., eds., *Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001)*, Williams College, Williamstown, MA, USA, June 28 - July 1, 2001, 609–616. Morgan Kaufmann.
- Zadrozny, B.; and Elkan, C. 2002. Transforming Classifier Scores into Accurate Multiclass Probability Estimates. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, 694–699. New York, NY, USA: Association for Computing Machinery. ISBN 158113567X.
- Zampieri, M.; Malmasi, S.; Nakov, P.; Rosenthal, S.; Farra, N.; and Kumar, R. 2019a. Predicting the Type and Target of Offensive Posts in Social Media. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 1415–1420. Minneapolis, Minnesota: Association for Computational Linguistics.
- Zampieri, M.; Malmasi, S.; Nakov, P.; Rosenthal, S.; Farra, N.; and Kumar, R. 2019b. SemEval-2019 Task 6: Identifying and Categorizing Offensive Language in Social Media (OffensEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, 75–86. Minneapolis, Minnesota, USA: Association for Computational Linguistics.
- Zampieri, M.; Nakov, P.; Rosenthal, S.; Atanasova, P.; Karadzhov, G.; Mubarak, H.; Derczynski, L.; Pitenis, Z.; and Çöltekin, Ç. 2020. SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020). In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, 1425–1447. Barcelona (online): International Committee for Computational Linguistics.
- Zampieri, N.; Illina, I.; and Fohr, D. 2023. Improving Hate Speech Detection with Self-Attention Mechanism and Multi-Task Learning. In *LTC'23 - 10th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*. Poznan, Poland.

Paper Checklist

1. For most authors...
 - (a) Would answering this research question advance science without violating social contracts, such as violating privacy norms, perpetuating unfair profiling, exacerbating the socio-economic divide, or implying disrespect to societies or cultures? **Yes.**
 - (b) Do your main claims in the abstract and introduction accurately reflect the paper's contributions and scope? **Yes.**
 - (c) Do you clarify how the proposed methodological approach is appropriate for the claims made? **Yes. See Section 6**
 - (d) Do you clarify what are possible artifacts in the data used, given population-specific distributions? **We use pre-existing datasets and do not have access to this information.**
 - (e) Did you describe the limitations of your work? **Yes. See Section 7.**
 - (f) Did you discuss any potential negative societal impacts of your work? **While our chosen application domain, hateful and offensive language, deals with content that is harmful by design, our method is a mere tool for analyzing and evaluating this phenomenon.**
 - (g) Did you discuss any potential misuse of your work? **We develop and evaluate quantification methods, which are a foundational tool with a large range of applications.**
 - (h) Did you describe steps taken to prevent or mitigate potential negative outcomes of the research, such as data and model documentation, data anonymization, responsible release, access control, and the reproducibility of findings? **We did not create any new corpora. The code for our experiments is available at <https://github.com/vodezhaw/icwsm2024/> mentioned in Section 1.**
 - (i) Have you read the ethics review guidelines and ensured that your paper conforms to them? **Yes.**
2. Additionally, if your study involves hypotheses testing...
 - (a) Did you clearly state the assumptions underlying all theoretical results? **We do not provide any theoretical results.**
 - (b) Have you provided justifications for all theoretical results? **We do not provide any theoretical results.**
 - (c) Did you discuss competing hypotheses or theories that might challenge or complement your theoretical results? **We do not provide any theoretical results.**
 - (d) Have you considered alternative mechanisms or explanations that might account for the same outcomes observed in your study? **No.**
 - (e) Did you address potential biases or limitations in your theoretical framework? **Limitations are discussed in Section 7.**
 - (f) Have you related your theoretical results to the existing literature in social science? **We do not provide any theoretical results.**
 - (g) Did you discuss the implications of your theoretical results for policy, practice, or further research in the social science domain? **We do not provide any theoretical results.**
3. Additionally, if you are including theoretical proofs...
 - (a) Did you state the full set of assumptions of all theoretical results? **We do not provide any theoretical results.**
 - (b) Did you include complete proofs of all theoretical results? **We do not provide any theoretical results.**
4. Additionally, if you ran machine learning experiments...
 - (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? **The URL (<https://github.com/vodezhaw/icwsm2024/>) is specified in Section 1.**
 - (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? **Yes. See Section 5.**
 - (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? **In Section 6 we give mean, median, and 95th percentile performances.**
 - (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? **No. Our experiments were not extraordinarily resource intensive.**
 - (e) Do you justify how the proposed evaluation is sufficient and appropriate to the claims made? **Yes. See Section 6.**
 - (f) Do you discuss what is “the cost” of misclassification and fault (in)tolerance? **This is somewhat dependent on the end-user's application domain. We point out the risks of bad prevalence estimation in Section 7.**
5. Additionally, if you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
 - (a) If your work uses existing assets, did you cite the creators? **See Section 5.1 for data and Sections 5.2 and 3 for models.**
 - (b) Did you mention the license of the assets? **We used existing datasets that were used for shared tasks. They are all intended to be used for research.**
 - (c) Did you include any new assets in the supplemental material or as a URL? **The URL (<https://github.com/vodezhaw/icwsm2024/>) is specified in Section 1.**
 - (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? **We are working with pre-existing datasets.**
 - (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? **We are working with pre-existing datasets.**
 - (f) If you are curating or releasing new datasets, did you discuss how you intend to make your datasets FAIR (see FORCE11 (2020))? **No new datasets were created.**

- (g) If you are curating or releasing new datasets, did you create a Datasheet for the Dataset (see Gebre et al. (2021))? No new datasets were created.
6. Additionally, if you used crowdsourcing or conducted research with human subjects...
- (a) Did you include the full text of instructions given to participants and screenshots? We use pre-existing data.
- (b) Did you describe any potential participant risks, with mentions of Institutional Review Board (IRB) approvals? We use pre-existing data.
- (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? We use pre-existing data.
- (d) Did you discuss how data is stored, shared, and de-identified? We use pre-existing data.

A Calibration Methods

	Random			Quantile		
	μ	Med.	95	μ	Med.	95
CPCC	0.073	0.049	0.211	0.085	0.059	0.275
CPCC-ISO	0.074	0.048	0.218	0.086	0.058	0.275
CPCC-HB10	0.086	0.054	0.273	0.096	0.064	0.297
CPCC-HB100	0.213	0.183	0.515	0.208	0.177	0.501

Table 10: *SAPE* of *CPCC* with different calibration methods. We report the mean (μ), median (Med.), and 95th percentile (95) of errors for all variants and sample selection methods.

In Section 4 we described the *CPCC* quantification method which relies on re-calibrating classifier scores. So far we have been using *Platt Scaling* (Platt et al. 1999) for this purpose. Here we repeat the main experiment described in Section 6.1 by considering additional calibration methods mentioned by Guo et al. (2017). For *CPCC-ISO*, we use *Isotonic Regression* (Zadrozny and Elkan 2002) instead of *Platt Scaling*. For *CPCC-HB10*, we use *Histogram Binning* (Zadrozny and Elkan 2001) with 10 bins and for *CPCC-HB100* we use 100 bins.

Table 10 shows the *SAPE* for the variants of *CPCC*. *CPCC-ISO* performs almost identically to *CPCC* with only slightly larger 95th percentile performance (0.218 versus 0.211). Both *Histogram Binning* methods perform worse. In particular, increasing the number of bins seems to decrease performance. For all variants, *random* sample selection outperforms *quantile* sample selection.

B Classifier Performance

In this section, we list all the sample-based performances of each of the 28 different classifiers. We report the ROC-AUC scores for the different classifiers. Note that the scores are highly diverse ranging from 0.469 to 0.989. Table 12 shows the scores for all *Electra* classifiers, Table 13 shows the scores for all *Twitter-RoBERTa* classifiers, Table 14 shows the scores for all *TD-IDF SVM* classifiers, and Table 15 shows the scores for the *Perspective API* classifier. We give an overview of the abbreviated dataset names in Table 11.

Name	Short Name
Davidson Hate & Offensive	DHO
Davidson Hate Only	DH
DGH	DGH
ETHOS	ETH
Ex Machina	ExM
HASOC 2019	HSC
HatEval 2019	HEv
Jigsaw	Jig
OLID	OL
SOLID	SOL
SWAD	SWD
WASSA	WSS

Table 11: Overview of the short names of the different datasets from Section 5.1 that are used in Tables 12, 13, 14, and 15.

	DHO	DH	DGH	Eth	ExM	HSC	HEv	Jig	OL	SOL	SWD	WSS
DHO	0.989	0.486	0.591	0.727	0.941	0.825	0.582	0.930	0.796	0.957	0.591	0.738
DH	0.442	0.894	0.639	0.769	0.852	0.720	0.621	0.800	0.711	0.696	0.620	0.670
DGH	0.718	0.679	0.901	0.886	0.573	0.610	0.737	0.554	0.487	0.531	0.485	0.510
ExM	0.937	0.587	0.552	0.763	0.980	0.857	0.603	0.966	0.885	0.974	0.631	0.773
HSC	0.876	0.623	0.543	0.733	0.938	0.833	0.598	0.904	0.835	0.913	0.716	0.737
HEv	0.920	0.457	0.657	0.830	0.899	0.741	0.663	0.890	0.759	0.857	0.642	0.759
Jig	0.913	0.546	0.543	0.775	0.978	0.855	0.601	0.972	0.885	0.974	0.600	0.788
OL	0.914	0.544	0.537	0.769	0.962	0.854	0.607	0.963	0.898	0.974	0.640	0.780
WSS	0.901	0.589	0.529	0.664	0.885	0.755	0.582	0.878	0.768	0.905	0.674	0.869

Table 12: ROC-AUC performance of *Electra* classifiers. Rows indicate the training set used, while columns indicate the target test set. We show the highest ROC-AUC achieved for each test set in bold. We abbreviated the dataset names, please refer to Table 11.

	DHO	DH	DGH	Eth	ExM	HSC	HEv	Jig	OL	SOL	SWD	WSS
DHO	0.984	0.469	0.558	0.703	0.941	0.816	0.588	0.928	0.820	0.968	0.592	0.699
DH	0.397	0.824	0.583	0.656	0.632	0.575	0.541	0.588	0.598	0.473	0.557	0.551
DGH	0.736	0.708	0.878	0.834	0.519	0.604	0.743	0.474	0.431	0.448	0.483	0.500
ExM	0.841	0.635	0.515	0.704	0.939	0.790	0.524	0.942	0.786	0.941	0.612	0.708
HSC	0.886	0.675	0.548	0.734	0.951	0.873	0.574	0.932	0.815	0.929	0.672	0.753
HEv	0.927	0.412	0.633	0.786	0.857	0.728	0.573	0.819	0.688	0.756	0.701	0.684
Jig	0.721	0.524	0.523	0.526	0.621	0.550	0.536	0.692	0.580	0.723	0.508	0.519
OL	0.906	0.524	0.506	0.760	0.948	0.858	0.607	0.954	0.890	0.970	0.622	0.836
WSS	0.933	0.655	0.522	0.622	0.845	0.649	0.534	0.870	0.628	0.861	0.641	0.862

Table 13: ROC-AUC performance of *Twitter-RoBERTa* classifiers. Rows indicate the training set used, while columns indicate the target test set. We show the highest ROC-AUC achieved for each test set in bold. We abbreviated the dataset names, please refer to Table 11.

	DHO	DH	DGH	Eth	ExM	HSC	HEv	Jig	OL	SOL	SWD	WSS
DHO	0.982	0.528	0.494	0.643	0.832	0.702	0.581	0.837	0.688	0.885	0.546	0.629
DH	0.366	0.839	0.555	0.670	0.733	0.676	0.548	0.709	0.592	0.619	0.564	0.549
DGH	0.483	0.547	0.636	0.612	0.384	0.435	0.559	0.399	0.491	0.296	0.515	0.514
ExM	0.879	0.604	0.503	0.705	0.957	0.820	0.580	0.953	0.767	0.955	0.619	0.709
HSC	0.639	0.538	0.421	0.568	0.645	0.734	0.556	0.567	0.567	0.818	0.592	0.590
HEv	0.858	0.395	0.541	0.696	0.683	0.618	0.625	0.692	0.622	0.703	0.577	0.602
Jig	0.875	0.611	0.507	0.713	0.965	0.833	0.576	0.963	0.796	0.962	0.600	0.739
OL	0.843	0.509	0.454	0.693	0.858	0.809	0.597	0.877	0.802	0.953	0.552	0.676
WSS	0.698	0.519	0.457	0.572	0.776	0.727	0.576	0.743	0.619	0.843	0.565	0.660

Table 14: ROC-AUC performance of *TF-IDF SVM* classifiers. Rows indicate the training set used, while columns indicate the target test set. We show the highest ROC-AUC achieved for each test set in bold. We abbreviated the dataset names, please refer to Table 11.

DHO	DH	DGH	Eth	ExM	HSC	HEv	Jig	OL	SOL	SWD	WSS
0.957	0.594	0.649	0.843	0.980	0.885	0.623	0.978	0.902	0.984	0.612	0.862

Table 15: ROC-AUC performance of *Perspective API* on our various test sets. We abbreviated the dataset names, please refer to Table 11.

Error-preserving Automatic Speech Recognition of Young English Learners' Language

Janick Michot¹, Manuela Hürlimann¹, Jan Deriu¹, Luzia Sauer².

Katsiaryna Mlynchik¹, Mark Cieliebak¹

¹ Zurich University of Applied Sciences, Winterthur

² Pädagogische Hochschule Zurich, Zurich

mict@zhaw.ch, hueu@zhaw.ch, deri@zhaw.ch, luzia.sauer@phzh.ch,

mlyn@zhaw.ch, ciel@zhaw.ch

Abstract

One of the central skills that language learners need to practice is speaking the language. Currently, students in school do not get enough speaking opportunities and lack conversational practice. Recent advances in speech technology and natural language processing allow for the creation of novel tools to practice their speaking skills. In this work, we tackle the first component of such a pipeline, namely, the automated speech recognition module (ASR), which faces a number of challenges: first, state-of-the-art ASR models are often trained on adult read-aloud data by native speakers and do not transfer well to young language learners' speech. Second, most ASR systems contain a powerful language model, which smooths out errors made by the speakers. To give corrective feedback, which is a crucial part of language learning, the ASR systems in our setting need to preserve the errors made by the language learners. In this work, we build an ASR system that satisfies these requirements: it works on spontaneous speech by young language learners and preserves their errors. For this, we collected a corpus containing around 85 hours of English audio spoken by learners in Switzerland from grades 4 to 6 on different language learning tasks, which we used to train an ASR model. Our experiments show that our model benefits from direct fine-tuning on children's voices and has a much higher error preservation rate than other models.

1 Introduction

Speaking is one of the core competencies to be developed in foreign language classes and the second most widely used skill in everyday-life communication (Hedge, 2001). For students to successfully acquire speaking competencies, they must be trained from an early stage in the language learning process and in a systematic manner. However, speech production is a highly complex process that is often not addressed adequately in class-

rooms. The main issue is that students often do not get enough speaking opportunities (Kleinschroth and Oldham, 2014; Grimm et al., 2015), and lack extended conversational practice (Pfenninger and Lendl, 2017). The recent advancements in both speech processing (Malik et al., 2021), and conversational dialogue systems (Deriu et al., 2021; Ni et al., 2023) provide an opportunity to increase the speaking practice of language learners using automated tools.

The work presented in this paper is part of a larger effort to develop an interactive, voice-driven chatbot with which learners can practice their interactive speaking skills. The bot is designed as a conversation partner that adjusts to the proficiency level and interests of the students and provides corrective feedback to support their language development.

One key issue is the automated speech recognition (ASR) module, which transcribes the utterances of the language learners into text to be processed in downstream tasks (e.g., speaker-error analysis, dialogue systems, inter alia). The focus of this work is to adapt the ASR module to handle children's speech in a language learning environment. The core challenge for the ASR system in this setting is not only to transcribe the speech but to make sure that the errors made by the language learners are transcribed faithfully. This is needed to provide language learners with corrective feedback, which is a key component of foreign language development. It prompts learners to notice errors and is likely to lead to utterance repair, which, in turn, facilitates language development (Ellis, 2021). Our investigations showed that current state-of-the-art ASR models tend to correct the speakers' errors, which renders giving corrective feedback impossible.

The second challenge for the ASR system is handling spontaneous children's speech since most of these systems are trained on adult read-aloud error-

6444

free corpora recorded by native speakers (Panayotov et al., 2015; Ardila et al., 2020). Children's speech, especially spontaneous speech of language learners, differs significantly from read-aloud speech of native adult speakers (Shivakumar and Georgiou, 2020). Children's speech has a different range of sound frequencies (Potamianos and Narayanan, 2003), a high within-subjects variability (Gerosa et al., 2006) and a high inter-speaker variability in different age groups (Lee et al., 1999).

These challenges yield three research questions, which we address in this work:

1. How can we measure error preservation, i.e. the "verbatimness" of an ASR transcript?
2. How well do current pre-trained ASR systems perform on learners' spontaneous English productions, with respect to error preservation and in general?
3. Does fine-tuning pre-trained systems with data from young learners lead to improved error preservation in the ASR transcripts?

Contributions In order to answer these questions, we first collected a dataset of young learners in Swiss public schools speaking English, consisting of 85 hours of recordings corresponding to 45'004 individual utterances by 327 distinct speakers. We subsequently created verbatim transcriptions of these recordings, in which learner errors are annotated using specific symbols. This dataset can be accessed on HuggingFace¹, but the dataset files must be downloaded manually as described in Section 3.4.2 below. We next developed a metric for error preservation, called *Word-Based Error Preservation Rate (WEPR)*, which takes into account only those reference words that contain an error annotation. Using WEPR and standard ASR metrics, we compared 7 pre-trained ASR systems with a custom fine-tuned model². Our results show that a) there are large differences between the pre-trained models both in terms of standard metrics and in terms of WEPR and b) fine-tuning significantly improves error preservation of learners' speech. All related code can be accessed on GitHub³.

¹<https://huggingface.co/datasets/mict-zhaw/chall>

²https://huggingface.co/mict-zhaw/chall_wav2vec2_xlsr_300m

³https://github.com/mict-zhaw/chall_e2e_stt

2 Related Work

Children's Speech Corpora. Corpora of children's speech can be divided into two types: i) corpora for native speaking children intended for building virtual tutors for non-language subjects, ii) corpora for young language learners that support building virtual tutors for language learning.

The MyST Children's Speech Corpus (Pradhan et al., 2016; Ward et al., 2019) contains 499 hours of conversational speech (out of which 233 hours are manually transcribed) for a virtual tutor for science topics targeted at young English native speakers. The OGI Kids' Speech Corpus (Shobaki et al., 2000) contains spontaneous speech from 1100 American children from kindergarten through grade 10, mainly consisting of scripted speech in the form of words and utterances, and a small sample of spontaneous speech. The AusKidTalk corpus (Ahmed et al., 2021) contains speech from Australian children ages 3 to 12 consisting of single words, utterances, and narrative speech. Other, smaller, datasets of native speaking children are available for different purposes such as read-aloud support (Eskenazi, 1996) or general analysis of English children's speech (Lee et al., 1999; Hagen et al., 2003). For German, the KidsTalk corpus (Rumberg et al., 2022) contains 25 hours of transcribed continuous speech from children aged 3 to 11. All these corpora are devised for settings with native speakers.

For language learners, there are far fewer datasets of children's speech. The TLT-school collection (Gretter et al., 2020) aims at assessing the proficiency of 9- to 16-year old Italian native speakers in English and German. TLT was recorded with a pool of 3000 students, resulting in approximately 275h of English and 265h of German data, out of which 16h for English and 8h for German have been transcribed. The corpus closest to our dataset is the CALL corpus (Baur et al., 2018), consisting of English utterances by Swiss German second and third year learners, where the task is to label the correctness of each utterance. In total, the corpus contains 38k utterances of students interacting with an online dialogue system, where they receive various prompts to produce speech. Across a series of shared tasks, subsets of around 6k annotated utterances have been released. The setting differs significantly from ours as we are interested in spontaneous speech with transcriptions to train an ASR system which can automatically transcribe learners'

speech verbatim. [Batliner et al. \(2005\)](#) introduce a children's speech corpus containing 60 hours of children's speech aged 4-13 in a variety of languages, such as English, German, and Swedish, as well as English speech from German, Italian, and Swedish children. In general, there is only a limited amount of work investigating the effects of fine-tuning models on speech data of language learners to retain the speakers' errors. [Ma et al. \(2023\)](#) fine-tune Whisper to investigate its ability to retain hesitations, numbers, abbreviations, disfluencies, and incomplete words. Instead, we aim to preserve speaker errors in grammar, lexical choice, and pronunciation.

ASR for children's speech and language learners. The literature on ASR models for children's speech, especially for non-native language learners, is sparse. Most notably, [Lu et al. \(2022\)](#) investigated the performance of fine-tuning wav2vec 2.0 ([Baevski et al., 2020](#)) on children's speech (both native MyST and OGI), as well as non-native speech (TLT) compared to fine-tuning on adult-only data. The results show that ASR models trained on children's speech significantly outperform those models trained on adult-speech only, even in the case of non-native speakers. Similarly, [Shivakumar and Narayanan \(2022\)](#) investigated the impact of using children's data for fine-tuning ASR models. The conclusion is similar to [Lu et al. \(2022\)](#): adding children's data yields better performance; however, the performance of an adult ASR model on adult data is higher than the performance of an ASR model trained and applied on children's data. While both [Lu et al. \(2022\)](#) and [Shivakumar and Narayanan \(2022\)](#) are interested in the overall performance in terms of WER, our work focuses on the preservation of errors made by non-native children.

3 Dataset: Spontaneous Speech of Young Learners of English

We now describe the dataset that we collected for the purpose of this research. It contains 85 hours of audio recordings of spontaneous speech by young Swiss learners of English. Each recording is paired with a verbatim transcript that contains error annotations.

3.1 Audio Recording

The recording setup was designed such that the collected speech resembled the kind of conversa-

tions intended for the learners to hold with the chatbot. We used playful and engaging activities targeted to elicit extended authentic communication from young learners. Activities included role plays with problem-solving components (e.g. 'going shopping for a school trip'), guessing games (e.g. riddles), TV interviews with imaginary characters and asking/answering personal questions (e.g. 'if you could go into space, what would you take with you?'). All activities were piloted with a grade 4 class and maintained, adjusted (to yield more data) or rejected (e.g. because the task led to students communicating non-verbally and/or with much noise) for the main data collection period. To support learners, each activity further included visual and language support (e.g. cartoon characters they could choose from, sample dialogues, language chunks) as well as a preparation phase during which the students could familiarise themselves with the tasks by use of example sentences and model dialogues.⁴

Speaker recruitment and consent After receiving permission to collect audio data with minors from key government institutions that act as ethics review boards in Switzerland concerning research with schools and their learners, we recruited 20 primary school teachers interested in participating in our project with their classes (via personal and university networks, newsletters and direct contact with schools). Participation was entirely voluntary and could be withdrawn at any time. Participation further necessitated the approval of the school principal and the written consent of each student's legal caretaker.⁵

In the span of 9 months (March-November 2023), 337 primary school students aged 9 to 14 years (4th to 6th graders) enrolled in 8 different schools in German-speaking Switzerland performed our activities in pairs, trios or alone (if necessary) in three different settings: at school recorded by project members; on the university campus recorded by project members and student assistants; and at school recorded by teachers and sent to us via safe weblinks. For reasons of practicability/feasibility (i.e., to respect teachers' tight schedules, time, and finances), the corpus was not annotated for CEFR levels, but according to the Swiss curriculum LP21, it should reflect performance at the A1 and A2 levels (English Ba-

⁴The descriptions of the speaking activities is provided in the repository.

⁵We share the consent forms in our repository.

sic Users). Some participants, including native-speaking children, performed beyond these levels. School principals, teachers, and students were not remunerated for their participation but received small tokens of appreciation, such as flowers and chocolates.

Metadata Each recording is associated with the following metadata:

- School area code: an integer between 1 and 8 (inclusive)
- School grade of the speakers: 4gr, 5gr, 6gr as well as combinations (4/5gr, 5/6gr, 4/6gr)
- Recording Device
- Recording Application
- Speaking activities
- Background Noise: a boolean indicating whether background noise is audible in the recording (set manually by project members).

3.2 Transcription and Error Annotation

The transcription of our voice data was outsourced to a transcription agency. Services included both the transcription of the voice data and the annotation of lexical, grammatical and pronunciation errors, as well as usage of German words. We developed a comprehensive data transcription guideline for the transcription agency which was first piloted on a small number of transcripts and then adjusted where necessary. Transcription guidelines included information about spelling conventions (British English), the frequency and nature of timestamps (start and end time of each word, in milliseconds), error codes (@! for errors of any kind and @g for German words) and disfluency markers (e.g. a hyphen "-" for verbatim repetitions, such as 'he's – he's really tall'). The complete transcription guidelines are provided in the supplementary material of this paper.

3.3 Data Aggregation and Filtering

The recording stage resulted in 1039 audio recordings. Of these, 23 were removed due to missing metadata or missing/retracted consent, so a total of 1016 recordings and their associated metadata and transcriptions were available for our experiments.

These recordings were split into individual utterances by a single speaker using the word-level timestamps provided in the transcripts, resulting in 49'608 utterances. We removed utterances shorter

than 0.5 seconds and utterances attributed to adults (e.g. short interventions by teachers), creating a final dataset of 45'004 utterances corresponding to 85 hours of audio. Each utterance was paired with its reference transcription and metadata.

3.4 Final Dataset

The final dataset contains 45'004 utterances by 327 distinct speakers. Figure 1 shows the number of recordings and audio duration by school grades and school area codes. Almost half the data in terms of both utterances and hours comes from 6th graders, while the other half is split among the other grades. The dataset contains 485,770 tokens and 10,203 distinct types. There are 14,396 error-annotated tokens with 2,004 underlying types. Thus, our data contains a large amount of tokens and a relatively large amount of token diversity.

The length distribution is shown in Figure 2. It can be seen that most utterances are between 0.5 and 20 seconds long.

3.4.1 Data Folds

For the experiments in this paper, we split the dataset into five distinct folds of similar duration (about 16h each), where each class (and therefore also each speaker) occurs in only one fold. To simulate the use case of the ASR system being confronted with a new class of learners, each fold contains data from a mix of grades. Figure 3 visualises the duration and grade distribution of each fold.

3.4.2 Data Availability

The dataset that we collected contains sensitive data of minors and thus cannot be shared publicly. The data can, however, be accessed as part of a joint project with one or several of the original project partners, subject to a collaboration agreement. Before sharing, all transcripts will undergo complete anonymisation so that any names and other personal information are removed.

4 Error-Preserving Automatic Speech Recognition

This section presents the metrics used for measuring error preservation and evaluating systems (Section 4.1), as well as the approaches to comparing pre-trained ASR systems (Section 4.2) and to fine-tuning existing systems using our learner dataset (Section 4.3). The qualitative results are presented

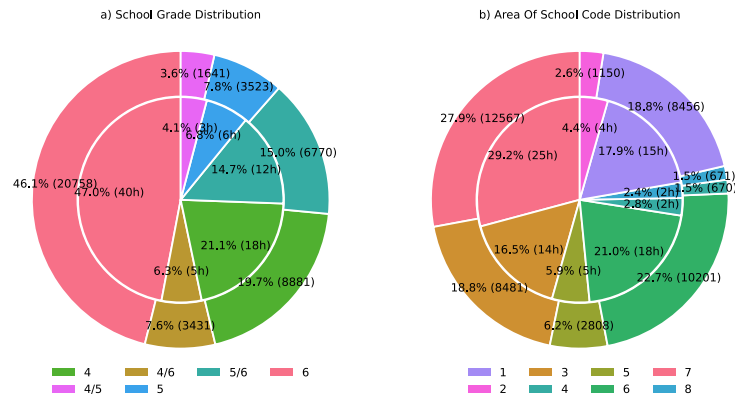


Figure 1: Number of utterances (outer ring) and audio hours (inner ring) by school grade (a) and school area code (b).

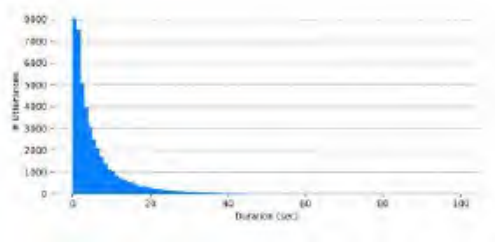


Figure 2: Distribution of utterance lengths.

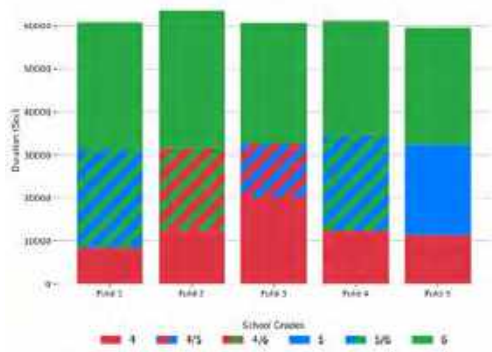


Figure 3: Duration and grade distribution of the data folds.

and discussed in Section 4.4, and a qualitative evaluation is shared in 4.5.

4.1 Metrics

In order to measure error preservation, we use the error annotations that were manually added to each utterance (cp. Section 3.2) and a custom phonetic word-level alignment algorithm. This algorithm aligns two or more sequences (e.g., a reference and one or multiple hypotheses), identifying matches,

substitutions (S), insertions (I), and deletions (D) at the word level. Our metric, WEPR (Word-Based Error Preservation Rate), considers only those word pairs where the reference word contains an error annotation. WEPR is calculated according to equation 1: $\mathcal{A}$ is the set of annotations that are considered (e.g. $\mathcal{A} = \{@!, @g\}$), $\mathcal{S}$ and $\mathcal{D}$ are the number of substitutions and deletions, respectively, where the reference word contains an error annotation, and $\mathcal{N}$ is the total number of reference words that contain an error annotation.

$$WEPR(\mathcal{A}) = \frac{(\mathcal{S} + \mathcal{D})}{\mathcal{N}} \quad (1)$$

In addition to WEPR, we also compute the following general ASR metrics using all words in the utterance: Word Error Rate (WER)⁶, Character Error Rate (CER)⁷, and character n-gram F-Score (chrF)⁸ (Popović, 2015).

We evaluate all models on our dataset's five folds (cp. Section 3.4.1) and report for each model the mean and standard deviation across all folds.

For evaluation, all texts are normalised using the Whisper normalizer for English⁹. Normalizing texts can mitigate the impact of disfluencies and non-standard linguistic forms, common in non-native and children's speech. This allows for a more accurate comparison between different ASR models, as it aligns the hypothesis and reference

⁶<https://github.com/huggingface/evaluate/blob/main/metrics/wer/wer.py>

⁷<https://github.com/huggingface/evaluate/blob/main/metrics/cer/cer.py>

⁸https://www.nltk.org/api/nltk.translate.chrf_score.html#nltk.translate.chrf_score.corpus_chrf

⁹<https://github.com/openai/whisper/blob/main/whisper/normalizers/english.py>

texts more closely.

Table 1 demonstrates text normalization and WEPR calculation using a Whisper Large model prediction and a modified normalizer. The adjustments to the Whisper normalizer maintain contractions in references, ensuring the integrity of annotated words (e.g., it's@!). For error preservation assessment, phonetic alignment is performed between the normalized prediction (NP) and a reference without error annotations (NR2). This removal is essential because the alignment algorithm would otherwise misidentify annotated words. The normalized reference with annotations (NR1) is then used to identify word pairs in the alignment results that are relevant for WEPR calculation. The Whisper Normalizer's tendency to increase the similarity between predictions and references helps in building and classifying these word-level pairs.

4.2 Pre-trained ASR Systems

We compare the performance of state-of-the-art ASR systems trained on datasets of adult English speakers. For this, we select seven different models, four based on a CTC decoding strategy, and three based on an encoder-decoder architecture. Our hypothesis is that CTC models are better at preserving speaker-errors as they do not rely on a language model, which potentially corrects such errors. Therefore, we do not use a n-gram language model during the CTC decoding phase, which is usually added for better WER performance. For the CTC-based models, we use the original Wav2Vec 2.0 large and base models (Baeovski et al., 2020) fine-tuned on 960h of Librispeech (Panayotov et al., 2015) (English adult read-aloud data). We also use the fine-tuned Wav2Vec 2.0 models provided by Grosman (2021, 2022), which are based on the XLSR pretraining (Babu et al., 2021), and were fine-tuned on the CommonVoice 6.1 data (Ardila et al., 2020) consisting of approximately 2100 hours of English adult read-aloud data. For the encoder-decoder architecture, we used the Whisper medium, large, and large-v3 models provided by OpenAI (Radford et al., 2022).

4.3 Fine-tuning Pre-trained ASR Systems Using Learner Data

To evaluate the impact of fine-tuning, we fine-tune the Wav2Vec-XLSR-300M model¹⁰ (Babu et al.,

¹⁰Due to the high computational cost, we decided to use the 300M model instead of the 1B model.

2021) on our collected language learner data.

Data Preprocessing. For fine-tuning, we split longer utterances into chunks of a maximum of 12 seconds and removed trailing pauses. The transcripts were preprocessed as follows:

- Remove error annotations and other transcript conventions
- Convert to lowercase
- Standardise text (Remove text between brackets and parentheses. Standardise apostrophes by removing spaces before them. Remove commas between digits and periods not followed by numbers.)
- Clean and standardise whitespace
- Normalise/remove special characters.
- Transform numbers into words using *num2words*

Approach. We apply 5-fold cross-validation (cf. 3.4.1), that is, we train on four folds, and evaluate on the held-out fold. We trained each run on 6 nVidia Tesla V100 GPUs for 4000 steps using a learning rate of 3e-5, a per-device batch size of 14, and 15 gradient accumulation steps (for a total batch size of 1260, which corresponds to approx. 2 hours of audio per batch), and we used the 8-bit AdamW optimizer (Loshchilov and Hutter, 2017; Dettmers et al., 2021). Our fine-tuned model, called ChaLL-300M, is available on HuggingFace.²

4.4 Quantitative Results

Performance Metrics. The scores achieved by the different models are summarised in Table 2. Among the pre-trained models, *Whisper-Large* achieves the best overall WER and chrF scores. However, the best CER and WEPR scores were achieved by the *XLSR-1B* models fine-tuned on CommonVoice 6.1. This aligns with our expectations, as Whisper models are currently the most powerful ASR models, and we expected them to perform best in terms of WER. However, for our use-case, we are more interested in error preservation, thus, CTC-based models without language models are best for preserving the errors. The fine-tuning step on our dataset consisting of learner data yielded a significant boost in performance. It achieves the best WEPR score, which measures the error retention capability. The most comparable model in terms of number of parameters is the XLSR-53 model trained on adult read-aloud data. In comparison to this model, *ChaLL-300M*

Category	Content
R	Reference
P	Prediction
NR1	Normalized Reference (with @!)
NR2	Normalized Reference
NP	Normalized Prediction
S	Substitutions
D	Deletions
I	Insertions
C	Correct
WEPR	0.67

Table 1: WEPR calculation example using Whisper Large model’s prediction with texts normalized by a customized version of Whisper’s Text Normalizer retaining contractions. Substitutions, insertions, deletions, and correct words are derived from phonetic alignment between NR2 and NP, but only for words that are annotated in NR1.

System Name	#Param.	WER	CER	chrF	WEPR
Wav2Vec Base	95M	0.55 ± 0.02	0.34 ± 0.02	0.35 ± 0.02	0.57 ± 0.02
Wav2Vec Large	317M	0.49 ± 0.02	0.29 ± 0.01	0.41 ± 0.02	0.50 ± 0.02
XLSR-53 + CommonVoice 6.1	317M	0.38 ± 0.01	0.26 ± 0.01	0.59 ± 0.01	0.50 ± 0.03
XLSR-1B + CommonVoice 6.1	1B	0.31 ± 0.01	0.21 ± 0.01	0.61 ± 0.01	0.44 ± 0.03
Whisper Medium	769M	0.26 ± 0.02	0.20 ± 0.03	0.70 ± 0.02	0.46 ± 0.04
Whisper Large	1.5B	0.25 ± 0.02	0.19 ± 0.01	0.70 ± 0.01	0.47 ± 0.03
Whisper Large-v3	1.5B	0.30 ± 0.04	0.23 ± 0.03	0.70 ± 0.02	0.45 ± 0.03
ChALL-300M (ours)	300M	0.30 ± 0.01	0.16 ± 0.01	0.68 ± 0.01	0.38 ± 0.03

Table 2: Results of the 5-fold evaluation. We report for each model the mean and standard deviation (mean±std) of the scores on each of the 5 folds. The bottom row shows the scores of our fine-tuned model.

TARGET	PREDICTION	CHALL-300M	WHISPER-LARGE	XLSR-1B
de@!	the	0.946	0.869	0.805
a@!	—	0.114	0.327	0.347
a@!	an	0.026	0.398	0.257
have@!	has	0.015	0.231	0.052
have@!	—	0.034	0.128	0.129
you@!	your	0.244	0.306	0.099
it’s@!	it	0.068	0.116	0.136
it’s@!	—	0.043	0.119	0.146
is@!	—	0.05	0.125	0.136
it’s@!	is	0.055	0.133	0.103
are@!	—	0.072	0.162	0.144
dis@!	this	0.854	0.83	0.717
it@!	—	0.094	0.311	0.193
he@!	—	0.175	0.254	0.356
de@!	—	0.029	0.123	0.143
the@!	—	0.046	0.24	0.183
in@!	—	0.027	0.127	0.107
you@!	—	0.077	0.113	0.117
i@!	—	0.133	0.248	0.294
on@!	—	0.019	0.129	0.105
Mean (n=20)		0.156	0.265	0.229

Table 3: System comparison on 20 most frequent incorrectly transcribed speaker-errors. For each system, the number indicates the fraction of cases in which the system incorrectly transcribes the error TARGET as PREDICTION (where "—" denotes deletion of TARGET). The lowest value of each row is set in boldface. The final row shows the mean across the 20 samples.

achieves an improvement of 8 points in WER and a 12-point improvement in WEPR. It is generally the case that larger models perform better. Thus, the interpretation of the results needs to factor this in. As most models are larger than ours, it becomes evident that fine-tuning on learner data increases the performance on this data in general, and the CTC

architecture yields a better out-of-the-box preservation of speaker-errors.

WEPR Analysis. To show in more detail the reduction in WEPR, we compare the handling of specific speaker errors. Table 3 shows the confusion for the 20 most frequent examples, that is, the cases where the ASR system corrects an error

TARGET	PREDICTION	CHALL-300M	WHISPER-LARGE	XLSR-1B
have@!	have	0.875	0.587	0.699
a@!	a	0.804	0.215	0.325
is@!	is	0.79	0.667	0.769
in@!	in	0.897	0.773	0.807
it's@!	it's	0.703	0.568	0.482
are@!	are	0.739	0.688	0.699
on@!	on	0.917	0.749	0.79
of@!	of	0.922	0.705	0.848
the@!	the	0.815	0.632	0.678
you@!	you	0.606	0.55	0.735
she@!	she	0.867	0.713	0.774
it@!	it	0.772	0.579	0.659
has@!	has	0.825	0.775	0.774
make@!	make	0.95	0.746	0.808
do@!	do	0.82	0.744	0.748
much@!	much	0.98	0.96	0.96
he@!	he	0.679	0.627	0.561
not@!	not	0.89	0.75	0.777
at@!	at	0.811	0.612	0.759
don't@!	don't	0.885	0.826	0.811
Mean (n=20)		0.827	0.673	0.723

Table 4: System comparison on 20 most frequent correctly preserved speaker-errors. For each system, the number indicates the fraction of cases in which the system correctly transcribes the error TARGET as PREDICTION. The highest value of each row is set in boldface. The final row shows the mean across the 20 samples.

	Utterance	Err. Type.
TARGET	Yeah. Uhm it's – It have a Lampe . Uhm you can –	has/have, German
CHALL300M	e uhm it's it's have a lampe you can	has/have, German
WHISPER-LARGE	it has a lamp	-
TARGET	(...) What you're rather be a (...) - able for fly or be invisible- invisible?	for/to
CHALL300M	would your reader be be aabble for fly or be invisible invisible	for/to
WHISPER-LARGE	would your reader be able to fly or be invisible	-
TARGET	Do you have a enemy?	a/an
CHALL300M	do you have a enemy	a/an
WHISPER-LARGE	do you have an enemy	-
TARGET	What do you favourite food?	do/is
CHALL300M	what do you favorite food	do/is
WHISPER-LARGE	what's your favorite food	-

Table 5: Manually selected examples.

it should have preserved. For each type of confusion, we report the rate at which it occurs. For instance, when the speaker mistakenly said "have" (denoted "have@!"), *ChaLL-300M* corrected it to "has" in 1.5% of cases, *Whisper-Large* corrected it in 23.1% of cases, and *XLSR-1B* in 12.9% of cases. Thus, *ChaLL-300M* preserved this particular kind of error the best. In total, it mistakenly corrected 15% of the 20 most frequent speaker-errors, while *Whisper-Large* corrected 26%, and *XLSR-1B* corrected 22.9%. It is interesting to note that two out of total three cases where *XLSR-1B* has the lowest rate of mis-correction is for pronunciation errors ("de@!" and "dis@!"). We also note that a majority of the most frequent unwanted error-corrections are deletions.

On the other hand, Table 4 shows the frequency at which the ASR systems correctly preserved the

errors made by the speakers. For instance, when the speaker mistakenly says "have" (denoted as "have@!"), then *ChaLL-300M* preserves this error in 87.5% of cases, while *Whisper-Large* preserves it in only 58.7% and *XLSR-1B* in only 69.9% of cases. In total, *ChaLL-300M* is able to preserve 82.7% of the 20 most frequent errors made by speakers, while *Whisper-Large* only preserved 67.3% of speaker errors and *XLSR-1B* preserved 72.3%.

Thus, *ChaLL-300M* displays a strong ability to preserve the errors made by speakers, which is crucial for the downstream task of providing automated corrective feedback.

4.5 Qualitative Results

Table 5 shows four manually selected examples, highlighting some errors which the best-performing pre-trained model, *Whisper-Large*, corrects, and

our model preserves. In the first example, it shows the error of using "have" instead of "has", as well as using the German pronunciation of the word "lamp" (i.e., "Lampe"). *Whisper-Large* corrects these errors, and creates a grammatically correct English utterance. The *ChaLL-300M* model preserves these errors as desired. The second error is a prepositional error, where the learner said "for fly" instead of "to fly". The *Chall-300M* model correctly preserved this error, while the language model used in *Whisper-Large* smoothed out the error. The third example is an error of the indefinite article: the learner used "a" instead of "an", which *ChaLL-300M* correctly preserved while *Whisper-Large* corrected the error. The final example contains the usage of the wrong verb "do" instead of "is", which again is correctly preserved by our model while *Whisper* corrects the error.

5 Conclusion and Outlook

Our work shows that state-of-the-art ASR systems have difficulties handling young learners' speech; furthermore, they tend to correct the errors made by the speakers, which makes the downstream identification of speaker errors and provision of corrective feedback impossible. Thus, we collected around 85 hours of children's language learner speech data, which we used to fine-tune a custom model. Our model outperforms all the others (including *Whisper-Large*) in terms of error preservation (Word-Based Error Preservation Rate, WEPR) and surpasses the English models of comparable size ($\approx 300M$ parameters) by a large margin in terms of Word Error Rate. Thus, our research shows the necessity of using targeted data (in this case, children who learn a foreign language) to fine-tune an ASR module, which is useful in downstream tasks. The focus of this work lies in a) investigating the utility of existing systems and b) creating an adequate ASR system that can be used as part of a language learning support tool to increase the students' speaking opportunities. As a next step, we will investigate how to enhance error preservation. For this, training larger models is the most straightforward approach. However, we also plan to train the ASR system jointly with error annotations. For this, we started the creation of more detailed error annotations. Initial results have shown that verbal errors are the largest error category for young learners of English in Switzerland (with about 22% of all errors)

, and within these, wrong subject-verb agreement is most frequent. Similarly, investigating how to handle frequent code-switching to German words or sentence fragments is an unsolved issue that needs to be addressed to improve downstream tasks. Even *Whisper-Large*, which can handle multiple languages in principle, did not perform well in detecting code-switching.

Finally, we aim to evaluate ASR models in the context of integrating them with a conversational agent and corrective feedback.

Acknowledgments

This work was supported by the Swiss Innovation Agency (Innosuisse) within the project "Towards a Voice-Based Chatbot for Language Learners (ChaLL)" [102.580.1 IP-ICT]. We thank the teachers and students who were part of the data collection for their efforts.

Limitations

While offering a unique tool for error-preserving ASR of young language learners, this work presents itself with a few limitations.

Limited Demographic. The dataset stems from a specific demographic of Swiss school children learning English in grades 4 to 6. An extension of the work would include language learners from other countries/with an academic language other than German/with a different language of instruction, or a larger range of ages. Thus, the transferability of our results must be confirmed with a different dataset.

Outsourcing Error Annotation. The outsourcing of transcription and error annotations always poses a risk of yielding erroneous data, since most transcribers are not trained in error annotation. We mitigated this risk by providing comprehensive guidelines and a steady exchange with the transcription agency. However, we plan to enhance the error annotations with a more detailed label set and annotators trained in this task.

Small Model. Due to the high computational cost of fine-tuning a 1B parameter model, we limited ourselves to fine-tuning the 300M parameter XLSR model. Most research indicates that the usage of larger models yields better results; thus, there is still potential in terms of increasing WER and WEPR. However, our results showed that even

a small model can preserve errors better than state-of-the-art pre-trained models, which was the main scope of this work.

No Performance Tuning. Since the scope of this work is to understand if the usage of young learners' speech data is beneficial for our purposes, we did not tune the performance of our model. That is, we did not perform any hyper-parameter tuning or any other methods to increase performance (e.g., joint prediction of errors using a language model). Thus, there is still a large margin of improvement using our dataset.

Data Availability. Since our data consists of children's spontaneous speech, we must ensure its protection. Thus, we cannot make it freely available. While we publicly release the models trained on the data, access to the transcripts and recordings can only be granted in the scope of a joint project, subject to a collaboration agreement.

Ethical Considerations

The main risks in this project have to do with data protection: all speakers are minors between 9 and 14 years of age, so their personal data must be very well safeguarded. Therefore, key government institutions approved the data collection before speakers were recruited, and informed consent was obtained from each speaker's legal caretaker (cp. details in Section 3.1). Consent forms entailed information about the nature of the project and data collection procedures, as well as a comprehensive description of the legal principles we followed to collect, use, and store voice data, transcripts, and annotations. The data protection measures we implemented for security and confidentiality were fully disclosed (e.g. password-protected documents, pseudonymisation, firewalls etc.) and risks to participants (e.g. potential voice recognition by project members) were outlined. Voice data and transcripts were pseudonymised by those project members who act as data owners before sharing them with other research partners and third parties. Third-party access to the collected data will be enabled in a closely controlled setting consisting of a joint project with a collaboration agreement.

Use of AI Assistants

ChatGPT was used to support the creation of some figures. No AI assistants were used to write the text of this paper.

References

- Beena Ahmed, Kirrie J. Ballard, Denis Burnham, Tharmakulasingam Sirojan, Hadi Mehmood, Dominique Estival, Elise Baker, Felicity Cox, Joanne Arciuli, Titia Benders, Katherine Demuth, Barbara Kelly, Chloé Diskin-Holdaway, Mostafa Shahin, Vidhyasaharan Sethu, Julien Epps, Chwee Beng Lee, and Eliathamby Ambikairajah. 2021. *AusKidTalk: An Auditory-Visual Corpus of 3- to 12-Year-Old Australian Children's Speech*. In *Proc. Interspeech 2021*, pages 3680–3684.
- R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber. 2020. Common voice: A massively-multilingual speech corpus. In *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020)*, pages 4211–4215.
- Arun Babu, Changhan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick von Platen, Yatharth Saraf, Juan Pino, et al. 2021. Xls-r: Self-supervised cross-lingual speech representation learning at scale. *arXiv preprint arXiv:2111.09296*.
- Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.
- Anton Batliner, Mats Blomberg, Shona D'Arcy, Daniel Elenius, Diego Giuliani, Matteo Gerosa, Christian Hacker, Martin Russell, Stefan Steidl, and Michael Wong. 2005. *The PF_STAR children's speech corpus*. In *Proc. Interspeech 2005*, pages 2761–2764.
- Claudia Baur, Andrew Caines, Cathy Chua, Johanna Gerlach, Mengjie Qian, Manny Rayner, Martin Russell, Helmer Strik, and Xizi Wei. 2018. *Overview of the 2018 spoken call shared task*. In *Proceedings of Interspeech 2018*, Interspeech, pages 2354–2358. ISCA. Interspeech 2018 ; Conference date: 02-09-2018 Through 06-09-2018.
- Jan Deriu, Alvaro Rodrigo, Arantxa Otegi, Guillermo Echegoyen, Sophie Rosset, Eneko Agirre, and Mark Cieliebak. 2021. Survey on evaluation methods for dialogue systems. *Artificial Intelligence Review*, 54:755–810.
- Tim Dettmers, Mike Lewis, Sam Shleifer, and Luke Zettlemoyer. 2021. 8-bit optimizers via block-wise quantization. *arXiv preprint arXiv:2110.02861*.
- Rod Ellis. 2021. Explicit and Implicit Oral Corrective Feedback. *The Cambridge Handbook of Corrective Feedback in Second Language Learning and Teaching*, pages 341–364.
- Maxine S Eskenazi. 1996. Kids: a database of children's speech. *The Journal of the Acoustical Society of America*, 100(4_Supplement):2759–2759.

- Matteo Gerosa, Diego Giuliani, and Shrikanth Narayanan. 2006. Acoustic analysis and automatic recognition of spontaneous children's speech. In *Ninth International Conference on Spoken Language Processing*.
- Roberto Gretter, Marco Matassoni, Stefano Bannò, and Falavigna Daniele. 2020. **TLT-school: a corpus of non native children speech**. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 378–385, Marseille, France. European Language Resources Association.
- Nancy Grimm, Michael Meyer, and Laurenz Volkman. 2015. *Teaching English*. Narr Francke Attempto Verlag.
- Jonatas Grosman. 2021. Fine-tuned XLSR-53 large model for speech recognition in English. <https://huggingface.co/jonatasgrosman/wav2vec2-large-xlsr-53-english>.
- Jonatas Grosman. 2022. Fine-tuned XLS-R 1B model for speech recognition in English. <https://huggingface.co/jonatasgrosman/wav2vec2-xls-r-1b-english>.
- Andreas Hagen, Bryan Pellom, and Ronald Cole. 2003. Children's speech recognition with application to interactive books and tutors. In *2003 IEEE Workshop on Automatic Speech Recognition and Understanding (IEEE Cat. No. 03EX721)*, pages 186–191. IEEE.
- Tricia Hedge. 2001. *Teaching and learning in the language classroom*, volume 106.
- Robert Kleinschroth and Pete Oldham. 2014. *Sprechkompetenz-Training im Englischunterricht 7-8: Lebensnahe Sprechanlässe und vielfältige Aufgaben (7. und 8. Klasse)*. Auer Verlag.
- Sungbok Lee, Alexandros Potamianos, and Shrikanth Narayanan. 1999. Acoustics of children's speech: Developmental changes of temporal and spectral parameters. *The Journal of the Acoustical Society of America*, 105(3):1455–1468.
- Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.
- Renee Lu, Mostafa Shahin, and Beena Ahmed. 2022. Improving children's speech recognition by fine-tuning self-supervised adult speech representations. *arXiv preprint arXiv:2211.07769*.
- Rao Ma, Mengjie Qian, Mark Gales, and Katherine M Knill. 2023. **Adapting an ASR Foundation Model for Spoken Language Assessment**. In *Proc. 9th Workshop on Speech and Language Technology in Education (SLaTE)*, pages 104–108.
- Mishaim Malik, Muhammad Kamran Malik, Khawar Mehmood, and Imran Makhdoom. 2021. Automatic speech recognition: a survey. *Multimedia Tools and Applications*, 80:9411–9457.
- Jinjie Ni, Tom Young, Vlad Pandelea, Fuzhao Xue, and Erik Cambria. 2023. Recent advances in deep learning based dialogue systems: A systematic survey. *Artificial intelligence review*, 56(4):3055–3155.
- Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an asr corpus based on public domain audio books. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 5206–5210. IEEE.
- Simone E. Pfenninger and Johanna Lendl. 2017. **Transitional woes: On the impact of l2 input continuity from primary to secondary school**. *Studies in Second Language Learning and Teaching*, 7(3):443–469.
- Maja Popović. 2015. chrF: character n-gram f-score for automatic mt evaluation. In *Proceedings of the tenth workshop on statistical machine translation*, pages 392–395.
- A. Potamianos and S. Narayanan. 2003. **Robust recognition of children's speech**. *IEEE Transactions on Speech and Audio Processing*, 11(6):603–616.
- Sameer Pradhan, Ron Cole, and Wayne Ward. 2016. **My science Tutor—Learning science with a conversational virtual tutor**. In *Proceedings of ACL-2016 System Demonstrations*, pages 121–126, Berlin, Germany. Association for Computational Linguistics.
- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. **Robust Speech Recognition via Large-Scale Weak Supervision**.
- Lars Rumberg, Christopher Gebauer, Hanna Ehlert, Maren Wallbaum, Lena Bornholt, Jörn Ostermann, and Ulrike Lüdtke. 2022. **kidsTALC: A Corpus of 3- to 11-year-old German Children's Connected Natural Speech**. In *Proc. Interspeech 2022*, pages 5160–5164.
- Prashanth Gurunath Shivakumar and Panayiotis Georgiou. 2020. **Transfer learning from adult to children for speech recognition: Evaluation, analysis and recommendations**. *Computer Speech & Language*, 63:101077.
- Prashanth Gurunath Shivakumar and Shrikanth Narayanan. 2022. **End-to-end neural systems for automatic children speech recognition: An empirical study**. *Computer Speech & Language*, 72:101289.
- Khalidoun Shobaki, John-Paul Hosom, and Ronald A. Cole. 2000. **The OGI kids² speech corpus and recognizers**. In *Proc. 6th International Conference on Spoken Language Processing (ICSLP 2000)*, pages vol. 4, 258–261.
- Wayne Ward, Ron Cole, and Sameer Pradhan. 2019. My science tutor and the myst corpus. *Boulder Learn. Inc.*

Originally published in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Bangkok, Thailand, August 2024, DOI: 10.18653/v1/2024.acl-long.243

Favi-Score: A Measure for Favoritism in Automated Preference Ratings for Generative AI Evaluation

Pius von Däniken and Jan Deriu and Don Tuggener and Mark Cieliebak

Centre for Artificial Intelligence
ZHAW School of Engineering
{vode, deri, tuge, ciel}@zhaw.ch

Abstract

Generative AI systems have become ubiquitous for all kinds of modalities, which makes the issue of the evaluation of such models more pressing. One popular approach is preference ratings, where the generated outputs of different systems are shown to evaluators who choose their preferences. In recent years the field shifted towards the development of automated (trained) metrics to assess generated outputs, which can be used to create preference ratings automatically. In this work, we investigate the evaluation of the metrics themselves, which currently rely on measuring the correlation to human judgments or computing sign accuracy scores.

These measures only assess how well the metric agrees with the human ratings. However, our research shows that this does not tell the whole story. Most metrics exhibit a disagreement with human system assessments which is often skewed in favor of particular text generation systems, exposing a degree of favoritism in automated metrics. This paper introduces a formal definition of favoritism in preference metrics, and derives the Favi-Score, which measures this phenomenon. In particular we show that favoritism is strongly related to errors in final system rankings. Thus, we propose that preference-based metrics ought to be evaluated on both sign accuracy scores and favoritism.

1 Introduction

With the rise of Generative AI for text, images, code, music, and other modalities, the question of how to evaluate the quality of the outputs of these systems is becoming more and more important. *Preference Ratings* have become a ubiquitous strategy for this task (Belz and Kow, 2010; Deriu et al., 2020; Stiennon et al., 2020; Freitag et al., 2021, 2022). The general idea is that the outputs of two generative systems are shown to a (typically human) rater who decides which output they prefer, or whether both outputs are of equal quality.

The ratings are then aggregated over a test set to compute in how many cases the outputs of one system were preferred over the other and whether the difference is significant.

Since human-based evaluations are expensive, there has been a growing movement towards training automated metrics to assess generated content. (Celikyilmaz et al., 2020; Deriu et al., 2021; Yeh et al., 2021). So instead of showing the outputs of two generative systems to human raters, one uses an automatic metric and hopes that the evaluation will resemble the ratings of the humans.¹ This raises the question of how these automated metrics themselves ought to be evaluated. Currently, the most popular method is measuring the correlation of the metrics' ratings to human judgments, or in the case of preference ratings, *sign-accuracy* on both the sample and system level is computed (i.e., on how many preference ratings do human and metrics agree, or on how many system-pairs do human and metrics agree) (Kocmi et al., 2021). For some tasks, such as machine translation, we have seen that according to these, metrics are steadily improving (Freitag et al., 2022).

While both measures are important in the assessment of automated metrics, they only count *how many* mistakes the metric typically makes, ignoring how these mistakes are distributed. That is, whether the mistakes are systematically skewed in favor of one generative system under investigation or another. To underscore the significance of this information, let us consider the assessment of two generative systems, denoted as π_a and π_b , using a metric that exhibits only a 10% error rate, which are all skewed towards π_a . This means that in 10% of cases, the metric's ratings will deviate from human judgments and in all cases the deviation is in favor π_a . Furthermore, assume that according

¹For this paper, we treat human ratings as gold-standard, but we are aware of the issues with human ratings (Amidei et al., 2018; Belz et al., 2021)

to humans π_a is slightly worse than π_b , then the favoritism of the metric will cause π_a to be rated as being better than π_b according to the metric.

Thus, the research question that we tackle in this work is how we can measure favoritism, and investigate its impact on evaluation.

Contributions. This work has three main contributions:

- We motivate and formally define the problem of favoritism in automated preference metrics for generative systems.
- We introduce the Favi-Score, an easy-to-compute score to quantify the favoritism exhibited by an automated preference metric in comparison to human judgments.
- We apply the Favi-Score to a variety of text generation tasks,² showcasing the interplay between the Favi-Score and existing measurements of preference-metric quality.

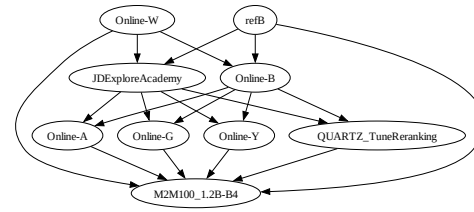
Our main finding is that favoritism causes mistakes in the rankings of systems according to the metric: even a metric with a high sign accuracy can lead to wrong evaluation if it has a strong favoritism. Or stated differently, a metric with low sign accuracy and no favoritism can still yield the correct ranking of systems.³

2 Impact of Favoritism on System Rankings

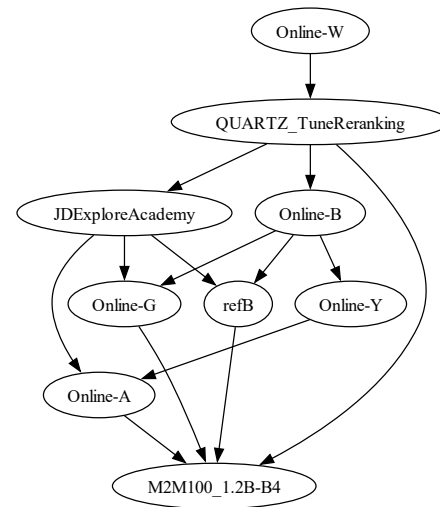
To demonstrate the need for the Favi-Score, we use the data of the WMT-22 metrics shared task (Freitag et al., 2022). This contains ratings for 9 machine translation systems. Based on this data, we produce one ranking using human ratings and one ranking using COMET-22 (Rei et al., 2022), a well-established automated metric for machine translation outputs. We visualize their assessments using directed acyclic graphs (DAGs), as depicted in Figure 1. For this, the preference ratings are computed for each pair of systems using the human ratings and the corresponding COMET-22 ratings. Then, we put an edge between two systems if one system is statistically significantly preferred to the other system according to a sign test (Coakley and Heise, 1996). For better visibility, we omit transitive edges. Note that the system-level agreement lies at 61.1%,

²Note that the formal definition of the Favi-Score could also be applied to other modalities. Thus, being agnostic to the task/domain at hand (e.g., music, images, or code generation).

³We provide an implementation of the Favi-Score: <https://github.com/vodezhaw/faviscore>



(a) Human



(b) COMET-22-refA

Figure 1: Visualization of the consequences of unfair metrics by comparing the human ranking to the COMET ranking. The ranking of a set of systems is depicted as a Directed Acyclic Graph, where an edge from system A to system B states that system A "wins" against system B. Here, a win is determined by a sign test (Coakley and Heise, 1996) at a 95% confidence threshold.

i.e., in 61.1% the edges of the COMET-22 DAG correspond to the ones of the human DAG.

We see in Figure 1a that according to humans, *ref-B* and *Online-W* are both at the top. However, in Figure 1b according to COMET-22, *ref-B* is ranked much worse, while *QUARTZ_TuneReranking* is disproportionately favored by COMET-22 compared to the human ranking. This is a direct consequence of favoritism.

Note that *only* considering the DAG does not reveal the whole story. For example, if a metric unduly favors a system that humans also rank highly, the overall ranking might not change. Yet, the favored system could appear to win by a much larger

margin than it deserves, exaggerating its performance. For a concrete example, if we consider the outcome percentages between *Online-W* and *JDExploreAcademy*, then under human evaluation, *Online-W* is rated better in 24.1% of cases and worse in 18.6% of cases (giving a margin of 5.5%). Under COMET-22, it is rated better in 57.5% of cases and worse in 42.2% (giving a margin of 15.3%). Even though *Online-W* is rated better than *JDExploreAcademy* in both cases, the perceived difference is much more pronounced under COMET-22, which is unfaithful to the human evaluation even though there is sign agreement. A concrete consequence is that, for example, a sign test will report much higher significance (lower p-value) for evaluation under COMET-22 than for human evaluation. This example shows the need for a mechanism that measures the inherent favoritism of a metric.

Informally, the Favi score is a relative measure between the human ratings and the metric ratings. *Instead of measuring how much the humans and metrics agree on their preference, it is a function of those samples where the humans and metrics disagree. The main idea is to count and weigh the cases in which the metric disproportionately assigns more samples in favor of one system than the human judges.* Thus, the Favi score yields a more fine-grained view compared to the DAG, as it also can measure the cases where there are no changes to the DAG, but one system is disproportionately favored (or disfavored) by the metric (see Examples in Appendix D). The Favi score can be used as a diagnostic tool to discover those cases. In the subsequent sections, we will formally introduce the Favi score.

3 Related Work

We are not aware of any prior work that defines or analyses favoritism in preference ratings. Most work on the analysis of metrics for text generation tasks is concerned with the overall performance, which is measured in terms of correlation to human judgments or sign accuracy (Deriu et al., 2021; Fabbri et al., 2021; Yeh et al., 2021; Freitag et al., 2022). The main conclusion of these analyses is that automated metrics do not yet match human ratings, and that the performance of the metrics vary depending on the domain and task. Moreover, there is a growing body of work that analyses individual metrics for generative tasks and their problems

(Schluter, 2017; Freitag et al., 2020; Amrhein and Sennrich, 2022; Hanna and Bojar, 2021, inter alia), but these studies do not compare or assess different metrics.

Novikova et al. (2017) go a step further and analyse specific weaknesses of various metrics for data-to-text. They showed that the metrics had difficulties distinguishing medium from good-quality outputs. Similarly, Kryscinski et al. (2019) showed that in the case of automated text summarization, automated metrics behave differently depending on whether the systems are extractive or abstractive. Thus, there is evidence in the literature that automated metrics are biased toward certain types of outputs and have difficulties with certain features. However, we are not aware of a fundamental analysis and discussion on the topic of structurally preferring the outputs of one system over others.

A different line of work is concerned with adversarial attacks on metrics. Sai et al. (2019) showed that ADEM (Lowe et al., 2017), a trained metric, is susceptible to specific changes in the inputs. Deriu et al. (2022) devised a method based on reinforcement learning to find nonsensical adversarial samples that elicit perfect scores from metrics for conversational dialogue systems. These findings indicate a lack of robustness and raise the question why certain outputs are preferred.

Finally, the last type of work related to ours is concerned with a more fundamental analysis of metrics. Wei and Jia (2021) apply the bias-variance-noise decomposition from Domingos (2000) to assess statistical bias of automated metrics used in the evaluation of machine translation and summarization. Their analysis produces a ranking of automated metrics based on their statistical bias, but it does not indicate the direction of the bias. Also, their approach does not investigate the relation between the bias strength and the systems under scrutiny. While they find that automated metrics for Natural Language Generation (NLG) are biased, we extend this notion and show that the strengths of the biases vary on the system level and affects system rankings. Chaganty et al. (2018) analyse the bias and variance of automated metrics. They find that automated metrics correlate poorly with human judgments, and that there is a correlation bias, i.e. the metrics often agree with humans in judging bad outputs but correlate poorly with humans on average or good quality outputs.

To summarize, there is strong evidence that automated metrics exhibit behavior which can benefit

or harm certain systems. To our knowledge, our work provides a definition and a measure for quantifying the amount of this benefit or harm for the first time.

4 Preference Ratings

In this section, we formalize the notion of a preference rating and define the confusion matrix which shows the amount of disagreement of two preference rating schemes (e.g. human judgments vs. an automated metric).

We first start by formally defining a *Generative System (GS)* as a function from an input space to an output space.

Definition 1 (Generative System (GS)). *Let $\mathcal{I}$ denote the set of all possible inputs to a GS, and $\mathcal{O}$ the set of all possible outputs. Then, a GS is a function π that takes an input and generates an output*

$$\pi : \mathcal{I} \rightarrow \mathcal{O} \quad (1)$$

Note that the input and output spaces are defined quite abstractly and are task-specific. For instance, in machine translation, the input space might be the set of utterances in the source language and the output space the set of utterances in the target language, whereas in news summarization, the input space would be the set of all news articles and the output is the set of all texts.

Next, we define the *Preference Rating* as a function of a triple of one input and two outputs to a sign.

Definition 2 (Preference Rating). *Let $\mathcal{I}$ denote the set of all possible inputs to a GS, and $\mathcal{O}$ the set of all possible outputs. Then, we define a preference rating as:*

$$\mathcal{R} : \mathcal{I} \times \mathcal{O} \times \mathcal{O} \rightarrow \{+, =, -\} \quad (2)$$

where $+$ denotes that the first output is preferred over the second, $=$ denotes that both outputs are of equal quality, and $-$ denotes that the second output is preferred.

Thus, for two generative systems π_1 and π_2 , $r[i] = \mathcal{R}(i, \pi_1(i), \pi_2(i))$ is the preference rating that compares the outputs of system π_1 and system π_2 for the input $i \in \mathcal{I}$. In the following, we denote a human-based preference rating as $r^H[i] = \mathcal{R}^H(i, \pi_1(i), \pi_2(i))$ and an automated metric-based preference rating as $r^A[i] = \mathcal{R}^A(i, \pi_1(i), \pi_2(i))$, respectively.

In order to run an evaluation and apply a preference rating $\mathcal{R}$ to decide whether π_1 performs better than π_2 , we first define the dataset that defines the evaluation setting.

Definition 3 (Evaluation Setting). *Given a test-set $\mathcal{T} \subseteq \mathcal{I}$, a pair of text generation systems π_1 and π_2 , and two preference ratings $\mathcal{R}^H$ and $\mathcal{R}^A$, we define the evaluation setting as:*

$$\mathcal{E} = \{i, \pi_1(i), \pi_2(i), r^H[i], r^A[i] | i \in \mathcal{T}\} \quad (3)$$

Thus, the evaluation setting $\mathcal{E}$ is composed of the inputs of the test set, the outputs of both TG systems, and the two preference ratings (e.g. according to humans and according to an automated metric).

For a given evaluation setting $\mathcal{E}$, we can compute the confusion matrix between $\mathcal{R}^A$ and $\mathcal{R}^H$.

Definition 4 (Confusion Matrix). *Given an evaluation setting $\mathcal{E}$, we define the confusion matrix C as*

$$C = \begin{pmatrix} C_{++} & C_{+=} & C_{+-} \\ C_{=+} & C_{==} & C_{=-} \\ C_{-+} & C_{-=} & C_{--} \end{pmatrix} \quad (4)$$

where $C_{mn} = \sum_{i \in \mathcal{T}} \mathbb{I}[r^H[i] = m \wedge r^A[i] = n]$ and $\mathbb{I}$ is the indicator function.

This corresponds to the usual confusion matrix known from classification settings, and the entries C_{mn} count how many times a rating of type m is classified as type n .

We call the cases where the automated rating disagrees with the human rating *errors*.

Definition 5 (Error). *Given an evaluation setting $\mathcal{E}$, $(i, \pi_1(i), \pi_2(i), r^H[i], r^A[i]) \in \mathcal{E}$ is an error (of the automated rating with respect to the human rating) if $r^A[i] \neq r^H[i]$.*

Definition 6 (Total Error). *Given an evaluation setting $\mathcal{E}$, the total error is given by $E = \sum_{i \in \mathcal{T}} \mathbb{I}[r^A[i] \neq r^H[i]]$.*

To decide which of the two systems is better, we define the *Outcome* of the evaluation, by counting the preferences.

Definition 7 (Outcome). *Given an evaluation setting $\mathcal{E}$, we define the outcome d as*

$$d = (d_+, d_-, d_0) \quad (5)$$

where $d_n = \sum_{i \in \mathcal{T}} \mathbb{I}[r^H[i] = n]$. We call $mar(d) = d_+ - d_-$ the outcome margin.

The outcome aggregates the preference ratings and counts how often one system is preferred over the other. For simplicity, we denote $\mathbf{d}$ as the outcome computed using the human ratings, and $\hat{\mathbf{d}}$ as the outcome according to the metric.

5 Measuring Favoritism

In this section, we define favoritism and derive a score to quantify it. Our view of favoritism is fundamentally relative, since it is based on comparing the behaviour of a metric with respect to the ground-truth provided by humans. In theory, one could compute favoritism compared to some other set of ground-truth labels (e.g., comparing two sets of human ratings). However, to keep our discussion simple, we treat human ratings as the unbiased ground truth, and leave deviations of this assumption to future work.

Our measure of favoritism, called Favi-Score, is based on two observations about errors of an automated rating with respect to the human rating. First, every error is necessarily in favor of a specific system. For example, changing a + to a =, is favoring π_2 . Second, not every error has the same severity. Switching a preference from one system to another (i.e., + to a -) is worse than classifying a draw as a preference (i.e., = to a -), since the former has a larger impact on the outcome margin than the latter. This means that the favoritism measure should depend both on the proportion of errors in favor of one system and the severity of those errors.

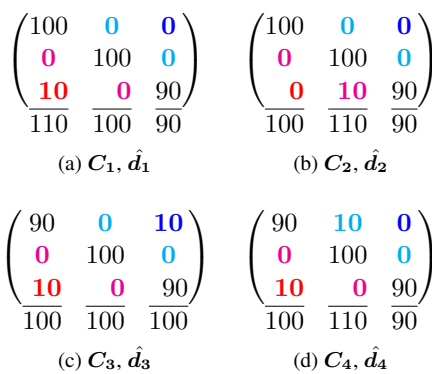


Figure 2: Example Confusion Matrices $\mathbf{C}$ and outcomes $\hat{\mathbf{d}}$ according to automated metrics for the human outcome $\mathbf{d} = (100, 100, 100)$.

To better illustrate the interplay of these criteria, let us consider the confusion matrices depicted in Figure 2 for an outcome of $\mathbf{d} = (100, 100, 100)$

alongside the corresponding outcome according to the metric $\hat{\mathbf{d}}$. We can see the error counts in the off-diagonal entries of $\mathbf{C}_i$, where the lower triangular sub-matrix, indicated in magenta and red, corresponds to errors in favor of π_1 and the upper triangular sub-matrix, indicated in cyan and blue, corresponds to errors in favor of π_2 . Let us first compare $\mathbf{C}_1$ and $\mathbf{C}_2$. In both cases there are 10 errors that are all in favor of π_1 . Nevertheless, $\mathbf{C}_1$ shows more favoritism since its errors are more severe, favoring π_1 more strongly. The consequence is visible in the outcome margin, where $\text{mar}(\hat{d}_1) = 110 - 90 = 20$ and $\text{mar}(\hat{d}_2) = 100 - 90 = 10$. $\mathbf{C}_3$ has more total errors, namely 20, but they are equally distributed in favor of each system and of equal severity. Therefore, $\mathbf{C}_3$ does not favor either system, which is also visible in the outcome margin $\text{mar}(\hat{d}_3) = 0$. Finally, $\mathbf{C}_4$ also has 20 errors distributed equally in favor of either system. But since the errors in favor of π_1 are more severe (i.e., mistaking + and -) than the ones in favor of π_2 (i.e., mistaking + and =), $\mathbf{C}_4$ favors π_1 overall. Thus, favoritism depends on how the errors are distributed, and how the severity of the mistakes are counted. The severity can be quantified in terms of how much they change the outcome according to the metric. Mistaking + for - leads to a change of 2 in the outcome margin, whereas mistaking + with = changes the outcome margin by 1. Thus, we weigh these mistakes accordingly and define the error cost:

Definition 8 (Directed Error Cost). *Let the following matrix be the error cost matrix:*

$$\mathbf{W} = \begin{pmatrix} 0 & -1 & -2 \\ 1 & 0 & -1 \\ 2 & 1 & 0 \end{pmatrix}$$

, where the magnitude of each entry denotes the cost for each mistake, and the sign denotes the direction of the favoritism.

Thus the cost of an error reflects its impact on the outcome margin, and the errors in favor of π_1 have a positive weight and errors in favor of π_2 have a negative weight. In cases where $m = n$ the cost $W_{mm} = 0$. Based on these observations, we will define a measure of favoritism, called Favi-Score Φ , that corresponds to the expected weight of the errors of an automated rating with respect to the human rating.

Definition 9 (Favi-Score). *Given an evaluation setting $\mathcal{E}$ and its confusion matrix $\mathbf{C}$ the Favi-Score*

$$C_5 = \begin{pmatrix} 360 & 180 & 60 \\ 20 & 40 & 40 \\ 90 & 90 & 120 \\ 470 & 310 & 220 \end{pmatrix}$$

Figure 3: Another example confusion matrix for illustration for $\mathbf{d} = (600, 100, 300)$.

is

$$\Phi(C) = \sum_{m,n} W_{mn} \frac{C_{mn}}{E}$$

The final score ranges from -2 to 2, where positive values indicate favoritism of π_1 and negative values indicate favoritism of π_2 .

The term $\frac{C_{mn}}{E}$ corresponds to the probability of a given type of error in cases where $m \neq n$, meaning $\frac{C_{mn}}{E} = P(r^H[i] = m \wedge r^A[i] = n | r^A[i] \neq r^H[i])$.

Thus, this brings us to the following interpretation of the Favi-Score:

The Favi-Score is the expected Error Cost.

There is an equivalent definition of the Favi-Score based on the outcome margins.

Lemma 1. *The following equality holds:*

$$\Phi(C) = \frac{\text{mar}(\hat{\mathbf{d}}) - \text{mar}(\mathbf{d})}{E} \quad (6)$$

Thus, the Favi-Score can also be interpreted as the average change in outcome margin per error.⁴

Coming back to our examples, the Favi-Scores for the confusion matrices in Figure 2 are therefore: $\Phi(C_1) = 2$, $\Phi(C_2) = 1$, $\Phi(C_3) = 0$, and $\Phi(C_4) = \frac{1}{2}$. In Figure 3 we show a more complex example. In this case more errors are in favor of π_2 (280 vs. 200), but more of the severe errors are in favor of π_1 (90 vs. 60). The overall Favi-Score $\Phi(C_5) = -0.104$ shows slight favor for π_2 . The score can be interpreted as each error contributing around 0.104 increase in margin in favor of π_2 .

6 Favi-Score vs. Sign Accuracy

In this section we explore the relation of the Favi-Score to sign accuracy. We show that the Favi-Score and sign accuracies measure different properties of a metric, and that the scores are complementary to each other.

First, we show the relationship between the Favi-Score and the system level sign accuracy. For this,

⁴The proof is in Appendix A.

we first define the system-level sign-accuracy using our notation.

Definition 10 (System-level sign accuracy). *Given $N = \binom{|\Pi|}{2}$ different pairs of systems. The formula for the system-level sign accuracy is given by:*

$$\frac{1}{N} \sum_{\pi^a, \pi^b} \mathbb{I}[\text{sgn}(d_+ - d_-) = \text{sgn}(\hat{d}_+ - \hat{d}_-)] \quad (7)$$

That is, the system level sign accuracy measures the number of times where the outcomes of the metric and the human evaluation agree on which system had more outputs of superior quality. The relationship between the Favi-Score and the System-level sign accuracy is that if there is no favoritism, the sign accuracy will be 1.

Lemma 2 (No System Level Sign Change). *If $\Phi(C) = 0$, that is, there is no favoritism, then the system-level sign will remain the same under the metric. That is: $\mathbb{I}[\text{sgn}(d_+ - d_-) = \text{sgn}(\hat{d}_+ - \hat{d}_-)] = 1$.*

Thus, a metric with no favoritism ensures that the outcome will lead to the same ranking. However, note that as discussed in Section 1, the converse might not hold, i.e., a system level sign accuracy of 1 does not imply that there is no favoritism. This brings us to the following insight:

Metrics with a low Favi-Score will tend to preserve the correct ranking regardless of their sign-accuracy.

The difference between the Favi-Score and the sample-level sign accuracy is apparent by considering its definition:

Definition 11 (Sample-level sign accuracy). *The sample-level sign accuracy is the fraction of samples, which are correctly predicted, i.e., the sum of the diagonal of C divided by the number of samples.*

$$\frac{1}{|\mathcal{T}|} \sum_{m \in \{+, -, \}} C_{m,m} = \frac{|\mathcal{T}| - E}{|\mathcal{T}|} \quad (8)$$

Thus, sign accuracy is a formula based on the diagonal entries of C , i.e., the fraction of samples where the metric and humans agree. Favoritism, on the other hand, is formulated via the difference of error types in favor of one system or the other, that is the upper and lower triangular sub-matrices. To illustrate this, we come back to our example in Figure 3, where $\Phi(C) = -0.104$, and the sample-level sign accuracy is 0.52. A different way to

showcase the difference is by assuming a fixed number of errors E , then the sign accuracy is 0 for $|\mathcal{T}| = E$, and approaches 1 for $|\mathcal{T}| \rightarrow \infty$, however, $|\mathcal{T}|$ does not influence $\Phi(C)$. Thus, sign accuracy tells us how often we can expect the human and metric to agree on a single sample, whereas favoritism tells us in which direction the mistakes are skewed and by how much.

7 Application of the Favi Score

For the empirical application, we require data in the form highlighted by the evaluation setting $\mathcal{E} = \{i, \pi_1(i), \pi_2(i), r^H[i], r^A[i] | i \in \mathcal{T}\}$. That is where we have a set of text generation systems with paired ratings from humans and automated metrics. For this, we use the data collected by Deriu et al. (2023), which consists of three different tasks (machine translation, summarization, and dialogue systems). We extended their dataset with data from the WMT-22 metrics shared task (see Appendix C for details). Here we provide an overview of the data (see Table 1). Note that for most metrics and human ratings there are either scalar or Likert outputs provided, which we transform to preference ratings by comparing the values (Appendix C).

Chatbot. The chatbot domain consists of the outputs of 5 different chatbots systems and one human reference automatically rated by 5 different metrics for the BlendedSkillTask (BST) dataset (Smith et al., 2020). The metrics consist of: DEB (Sai et al., 2020), GRADE (Huang et al., 2020), HolisticEval (Pang et al., 2020), MAUDE (Sinha et al., 2020), and USL-H (Phy et al., 2020). There are 50 human ratings for each pair of systems, yielding $|\mathcal{E}| = 50$ paired ratings.

Summarization. The summarization domain is based on the data of the SummEval framework (Fabbri et al., 2021), which provides the outputs of 16 different summarization tools on the CNN/DailyMail corpus (Nallapati et al., 2016) rated by 7 different automated metrics: BertScore (Zhang et al., 2019), BLANC (Vasilyev et al., 2020), CIDEr (Vedantam et al., 2015), Rouge-L (Lin, 2004), S3 (Peyrard et al., 2017), SummaQA (Scialom et al., 2019), and SUPERT (Gao et al., 2020). For 100 outputs, there are expert Likert ratings for each of the features: relevance, coherence, consistency, and fluency, thus $|\mathcal{E}| = 100$.

Machine Translation. For machine translation, we use the WMT-21 metrics task data (Freitag et al., 2021), and the WMT-22 metrics task data (Fre-

	Chatbot	SummEval	WMT21	WMT22
Data	BST	CNN/DM	News EN->DE	EN->DE
Metrics	5	7	4	5
TG Systems	6	16	11	9
$ \mathcal{E} $	50	100	500	1315

Table 1: Overview of Data Statistics. The ratings refer to the number of ratings available for each pair of TG systems.

itag et al., 2022). For WMT-21 we use the English to German language pair and the news domain where eight machine translation systems were evaluated, plus three human references. We used the four most prominent metrics: BleuRT (Selam et al., 2020), COMET (Rei et al., 2020), C-SPEC (Takahashi et al., 2021), and sentence-level BLEU (Papineni et al., 2002). There are 500 ratings by expert translators for each of the translation systems, thus, $|\mathcal{E}| = 500$. For WMT-22, we use the English to German language pair and selected eight machine translation systems plus one human reference, and five metrics. Namely, BLEU, BleuRT, COMET22 (Rei et al., 2022), UniTE (Wan et al., 2022), and METRICX_XL.⁵ We additionally added ChatGPT⁶ as a preference metric (see Appendix C.2). For 1315 samples, expert ratings were provided, thus $|\mathcal{E}| = 1315$.

7.1 Results & Discussion

Overall Results. We compare the Favi-Score to the sample- and system-level sign accuracy. Since the Favi-Score is computed for each pair of systems, we report the absolute average Favi-Score over each pair of systems alongside the standard-deviation, while for the two sign accuracy scores, we report the overall score. Figure 4 shows the three scores for each domain and metric. Note that for the Favi-Score (green) we also report the standard deviation (green area). First, we note that each metric exhibits a certain degree of favoritism, with some degree of variance with regards to the system pairs. The magnitude of the Favi-Score varies depending on the domain, for instance, machine-translation metrics exhibit a lower amount of favoritism, while the summarization metrics exhibit a larger amount as well as a larger variance.

System Pair Analysis. To get a better understanding of the interplay between the three measures, we compute the sign-accuracy on the system-pair level (for the system-level sign accuracy this means that we return whether the signs agree or

⁵No reference available.

⁶gpt-3.5-turbo-0613

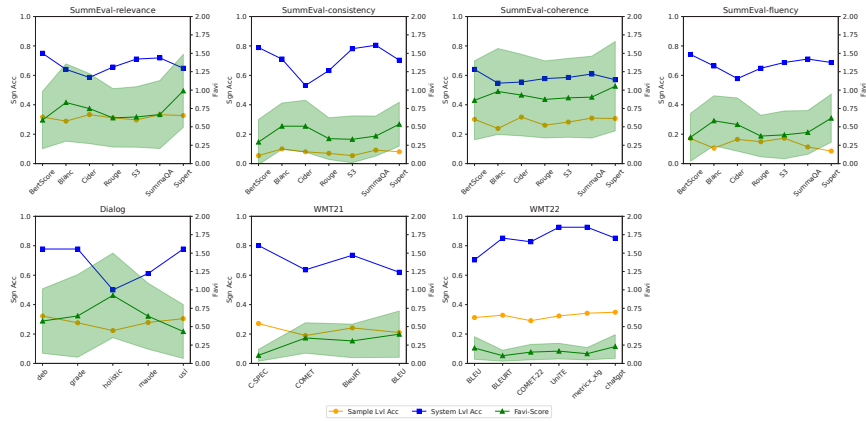
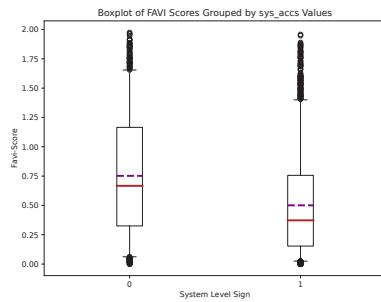
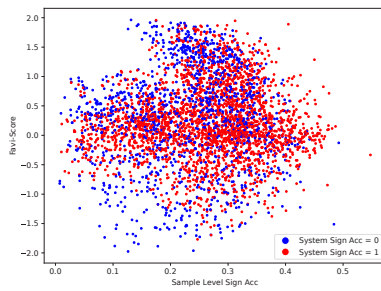


Figure 4: Visualization of the sample-level sign accuracy (orange), system-level sign accuracy (blue), and the average favi-score with a standard deviation (green).



(a) Favi vs System Level Sign Acc



(b) Favi vs Sample Level Sign Acc

Figure 5: The relationship between the Favi-score and both types of sign accuracy. In the upper plot, the boxplot show the distribution of absolute Favi-scores for the cases where the System Level sign agrees for a pair (1) and for the cases where it disagrees (0). The lower plot scatters the Favi-scores vs. the sample level accuracy, where the color showcases the system-level accuracy.

not). First, we investigate whether the system-level sign accuracy correlates to the Favi-Score. In Figure 5a, we show the Favi-Scores depending on

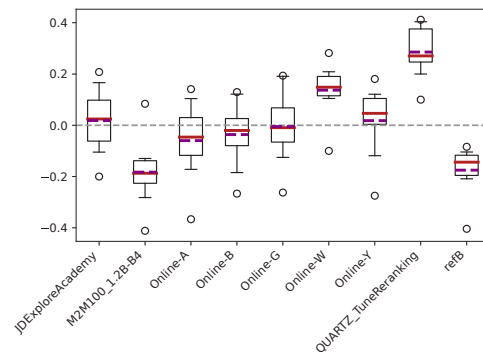


Figure 6: Box Plot of the Favi-Scores for each System for the COMET-22 metric.

whether the sign agrees on the system level (1) or not (0). We observe that system-pairs where the sign does not agree tend to have a larger Favi-Score, which is expected according to Lemma 2. In fact, there is a low-to-moderate inverse correlation between the two scores of Spearman's $\rho = -0.25$. The correlation of the Favi-Score to the sample-level sign accuracy is depicted in Figure 5b, which shows a small correlation (Spearman's $\rho = -0.01$). We note that very high sample-level accuracies are associated with smaller Favi-Scores.

Return to Motivating Example. To return to the motivating example from Section 2, we showcase the Favi-Scores for each system (one-vs-all) according to COMET-22 in Figure 6. That is we computed the Favi-Score for each pair of systems, and display for each system all Favi-Scores in terms of a box-plot. COMET-22 strongly disfavors *ref-B*

(i.e., the human reference), since the Favi-Score is negative for *ref-B*, no matter the system we compare against. This matches the DAG in Figure 1 in which *ref-B* is ranked significantly lower than in the human ranking. The *M2M100* system is also strongly disfavored. However, this is not visible from the DAGs alone, where it is ranked last, both according to COMET-22 and human judgements. This is a case where the Favi-Score uncovers a potentially undesired outcome that could go unnoticed otherwise, highlighting the utility of the Favi-Score as a diagnostic tool. Finally, we observe a strong favoritism with regards to the *QUARTZ* system, which is ranked significantly higher according to COMET-22. Thus, the Favi-Score is also useful as a diagnostic tool and provides an explanation for observed errors in system rankings. See Appendix D for more such examples.

The results of the application of the Favi-Score to evaluate metrics shows that it measures a different aspect of the metrics behaviour than the sign-accuracy scores. Thus, we advocate the usage of the Favi-Score alongside existing scores when a novel metric for generative systems is developed.

8 Conclusion

We motivated and defined the Favi-Score, a simple-to-implement and easy to interpret score, which measures how much an automated preference metric favors the outputs of a generative AI system. Our investigations show that the Favi-Score differs from the sign accuracy scores, and is to be used complementary to them. The empirical application of the Favi-Score to real-world setting reveals that all metrics exhibit some degree of favoritism, which needs to be accounted for when applying these metrics, to better interpret results. We envision that the Favi-Score is used as a diagnostic tool to understand which systems are favored by a metric.

Acknowledgments

This work was supported by the Swiss National Science Foundation within the project "Unified Model for Evaluation of Text Generation Systems (UniVal)" [200020_219819].

Limitations

Relative Measure. Our definition of favoritism is fundamentally relative, meaning we can only say

that some automated preference ratings display favoritism with respect to a specific set of human ratings.

Anchoring. This makes it crucial to prudently select the reference ratings that are used for the comparison. In this work, we treat human ratings as the gold standard reference despite their limitations (Amidei et al., 2018; Belz et al., 2021).

Comparisons. The Favi-Score is defined for a given evaluation setting $\mathcal{E}$, meaning for a fixed set of automated ratings, human ratings, and system pair. If we replace the the automated ratings, the resulting new Favi-Score is directly comparable. On the other hand, comparing the Favi-Score for one system pair to another, even for the same automated metric, means we have to assume a degree of consistency between reference ratings.

Alternative Definitions. Our definition of favoritism is narrowly defined based on (perceived) errors with respect to human ratings. In particular, we do not consider the content of either the inputs or outputs of various systems under test. Even within our narrow framework, one could imagine alternative definitions taking into account how a change in outcome affects significance tests.

Robustness. Throughout this work, we consider confusion matrices based on reasonably sized evaluation settings. For smaller settings the utility of the Favi-Score can be limited. For example, if there is only 1 error then the only possible values of the Favi-Score are ± 1 and ± 2 . This could potentially be mitigated by a full probabilistic treatment of the evaluation setting, which is outside the scope of this work.

Diagnostic. While we are convinced that the Favi-Score adds a useful diagnostic to a practitioner's toolbox, it does not provide any remedies. For example, to lower the Favi-Score, one can try to distribute errors more evenly, but it is not clear how this should be achieved in a concrete case.

References

- Jacopo Amidei, Paul Piwek, and Alistair Willis. 2018. [Rethinking the agreement in human evaluation tasks](#). In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3318–3329, Santa Fe, New Mexico, USA. Association for Computational Linguistics.
- Chantal Amrhein and Rico Sennrich. 2022. [Identifying weaknesses in machine translation metrics through minimum Bayes risk decoding: A case study for COMET](#). In *Proceedings of the 2nd Conference*

- of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 1125–1141, Online only. Association for Computational Linguistics.
- Anja Belz and Eric Kow. 2010. [Comparing rating scales and preference judgements in language evaluation](#). In *Proceedings of the 6th International Natural Language Generation Conference*. Association for Computational Linguistics.
- Anya Belz, Anastasia Shimorina, Shubham Agarwal, and Ehud Reiter. 2021. [The ReproGen shared task on reproducibility of human evaluations in NLG: Overview and results](#). In *Proceedings of the 14th International Conference on Natural Language Generation*, pages 249–258, Aberdeen, Scotland, UK. Association for Computational Linguistics.
- Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2020. Evaluation of text generation: A survey. *arXiv preprint arXiv:2006.14799*.
- Arun Chaganty, Stephen Mussmann, and Percy Liang. 2018. [The price of debiasing automatic metrics in natural language evaluation](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 643–653, Melbourne, Australia. Association for Computational Linguistics.
- Clint W. Coakley and Mark A. Heise. 1996. [Versions of the sign test in the presence of ties](#). *Biometrics*, 52(4):1242–1251.
- Jan Deriu, Alvaro Rodrigo, Arantxa Otegi, Guillermo Echegoyen, Sophie Rosset, Eneko Agirre, and Mark Cieliebak. 2021. Survey on evaluation methods for dialogue systems. *Artificial Intelligence Review*, 54(1):755–810.
- Jan Deriu, Don Tuggener, Pius von Däniken, Jon Ander Campos, Alvaro Rodrigo, Thiziri Belkacem, Aitor Soroa, Eneko Agirre, and Mark Cieliebak. 2020. [Spot the bot: A robust and efficient framework for the evaluation of conversational dialogue systems](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3971–3984, Online. Association for Computational Linguistics.
- Jan Deriu, Don Tuggener, Pius Von Däniken, and Mark Cieliebak. 2022. [Probing the robustness of trained metrics for conversational dialogue systems](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 750–761, Dublin, Ireland. Association for Computational Linguistics.
- Jan Deriu, Pius von Däniken, Don Tuggener, and Mark Cieliebak. 2023. [Correction of errors in preference ratings from automated metrics for text generation](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 6456–6474, Toronto, Canada. Association for Computational Linguistics.
- Pedro M. Domingos. 2000. A unified bias-variance decomposition and its applications.
- Alexander R. Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. [SummEval: Re-evaluating summarization evaluation](#). *Transactions of the Association for Computational Linguistics*, 9:391–409.
- Markus Freitag, David Grangier, and Isaac Caswell. 2020. [BLEU might be guilty but references are not innocent](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 61–71, Online. Association for Computational Linguistics.
- Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André F. T. Martins. 2022. [Results of WMT22 metrics shared task: Stop using BLEU – neural metrics are better and more robust](#). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 46–68, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, George Foster, Alon Lavie, and Ondřej Bojar. 2021. [Results of the WMT21 metrics shared task: Evaluating metrics with expert-based human evaluations on TED and news domain](#). In *Proceedings of the Sixth Conference on Machine Translation*, pages 733–774, Online. Association for Computational Linguistics.
- Yang Gao, Wei Zhao, and Steffen Eger. 2020. [SUPERT: Towards new frontiers in unsupervised evaluation metrics for multi-document summarization](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1347–1354, Online. Association for Computational Linguistics.
- Michael Hanna and Ondřej Bojar. 2021. [A fine-grained analysis of BERTScore](#). In *Proceedings of the Sixth Conference on Machine Translation*, pages 507–517, Online. Association for Computational Linguistics.
- Lishan Huang, Zheng Ye, Jinghui Qin, Liang Lin, and Xiaodan Liang. 2020. [GRADE: Automatic graph-enhanced coherence metric for evaluating open-domain dialogue systems](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9230–9240, Online. Association for Computational Linguistics.
- Tom Kocmi, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes. 2021. [To ship or not to ship: An extensive evaluation of automatic metrics for machine translation](#). In *Proceedings of the Sixth Conference on Machine Translation*, pages 478–494, Online. Association for Computational Linguistics.
- Klaus Krippendorff. 1970. [Estimating the reliability, systematic error and random error of interval](#)

- data. *Educational and Psychological Measurement*, 30(1):61–70.
- Wojciech Kryscinski, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. [Neural text summarization: A critical evaluation](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 540–551, Hong Kong, China. Association for Computational Linguistics.
- Chin-Yew Lin. 2004. [ROUGE: A Package for Automatic Evaluation of Summaries](#). In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.
- Ryan Lowe, Michael Noseworthy, Iulian Vlad Serban, Nicolas Angelard-Gontier, Yoshua Bengio, and Joelle Pineau. 2017. [Towards an automatic Turing test: Learning to evaluate dialogue responses](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1116–1126, Vancouver, Canada. Association for Computational Linguistics.
- Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gulçehre, and Bing Xiang. 2016. [Abstractive text summarization using sequence-to-sequence RNNs and beyond](#). In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.
- Jekaterina Novikova, Ondřej Dušek, Amanda Cercas Curry, and Verena Rieser. 2017. [Why we need new evaluation metrics for NLG](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2241–2252, Copenhagen, Denmark. Association for Computational Linguistics.
- OpenAI. 2023. GPT-3.5 Turbo (0613). <https://platform.openai.com/docs/models/gpt-3-5-turbo>. Accessed: 2024-01-31.
- OpenAI. 2023. OpenAI API. <https://platform.openai.com/docs/guides/function-calling>. Accessed: 2024-01-31.
- Bo Pang, Erik Nijkamp, Wenjuan Han, Linqi Zhou, Yixian Liu, and Kewei Tu. 2020. [Towards holistic and automatic evaluation of open-domain dialogue generation](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3619–3629, Online. Association for Computational Linguistics.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. [Bleu: a Method for Automatic Evaluation of Machine Translation](#). In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.
- Maxime Peyrard, Teresa Botschen, and Iryna Gurevych. 2017. [Learning to score system summaries for better content selection evaluation](#). In *Proceedings of the Workshop on New Frontiers in Summarization*, pages 74–84, Copenhagen, Denmark. Association for Computational Linguistics.
- Vitou Phy, Yang Zhao, and Akiko Aizawa. 2020. [Deconstruct to reconstruct a configurable evaluation metric for open-domain dialogue systems](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4164–4178, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. 2022. [COMET-22: Unbabel-IST 2022 submission for the metrics shared task](#). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 578–585, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. [COMET: A neural framework for MT evaluation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.
- Ananya B. Sai, Mithun Das Gupta, Mitesh M. Khapra, and Mukundhan Srinivasan. 2019. [Re-evaluating adem: A deeper look at scoring dialogue responses](#). *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6220–6227.
- Ananya B. Sai, Akash Kumar Mohankumar, Siddhartha Arora, and Mitesh M. Khapra. 2020. [Improving Dialog Evaluation with a Multi-reference Adversarial Dataset and Large Scale Pretraining](#). *Transactions of the Association for Computational Linguistics*, 8:810–827.
- Natalie Schluter. 2017. [The limits of automatic summarisation according to ROUGE](#). In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 41–45, Valencia, Spain. Association for Computational Linguistics.
- Thomas Scialom, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. 2019. [Answers unite! unsupervised metrics for reinforced summarization models](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3246–3256, Hong Kong, China. Association for Computational Linguistics.
- Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. [BLEURT: Learning robust metrics for text generation](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages

- 7881–7892, Online. Association for Computational Linguistics.
- Koustuv Sinha, Prasanna Parthasarathi, Jasmine Wang, Ryan Lowe, William L. Hamilton, and Joelle Pineau. 2020. [Learning an unreferenced metric for online dialogue evaluation](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2430–2441, Online. Association for Computational Linguistics.
- Eric Michael Smith, Mary Williamson, Kurt Shuster, Jason Weston, and Y-Lan Boureau. 2020. [Can you put it all together: Evaluating conversational agents' ability to blend skills](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2021–2030, Online. Association for Computational Linguistics.
- Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. [Learning to summarize with human feedback](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 3008–3021. Curran Associates, Inc.
- Kosuke Takahashi, Yoichi Ishibashi, Katsuhito Sudoh, and Satoshi Nakamura. 2021. [Multilingual machine translation evaluation metrics fine-tuned on pseudo-negative examples for WMT 2021 metrics task](#). In *Proceedings of the Sixth Conference on Machine Translation*, pages 1049–1052, Online. Association for Computational Linguistics.
- Oleg Vasilyev, Vedant Dharnidharka, and John Bohannon. 2020. [Fill in the BLANC: Human-free quality estimation of document summaries](#). In *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems*, pages 11–20, Online. Association for Computational Linguistics.
- Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Yu Wan, Dayiheng Liu, Baosong Yang, Haibo Zhang, Boxing Chen, Derek Wong, and Lidia Chao. 2022. [UniTE: Unified translation evaluation](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8117–8127, Dublin, Ireland. Association for Computational Linguistics.
- Yidong Wang, Zhuohao Yu, Zhengran Zeng, Linyi Yang, Cunxiang Wang, Hao Chen, Chaoya Jiang, Rui Xie, Jindong Wang, Xing Xie, Wei Ye, Shikun Zhang, and Yue Zhang. 2024. Pandalm: An automatic evaluation benchmark for llm instruction tuning optimization.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. [Chain-of-thought prompt elicits reasoning in large language models](#). In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.
- Johnny Tian-Zheng Wei and Robin Jia. 2021. The statistical advantage of automatic nlg metrics at the system level. In *Annual Meeting of the Association for Computational Linguistics*.
- Yi-Ting Yeh, Maxine Eskenazi, and Shikib Mehri. 2021. [A comprehensive assessment of dialog evaluation metrics](#). In *The First Workshop on Evaluations and Assessments of Neural Conversation Systems*, pages 15–33, Online. Association for Computational Linguistics.
- Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

A Proof of Equal Definition

Here, we prove Lemma 1, which states:

$$\sum_{m,n} W_{mn} \frac{C_{mn}}{E} = \frac{\text{mar}(\hat{d}) - \text{mar}(d)}{E} \quad (9)$$

We start the proof by expanding both sides of the equation. We start with the left side. Let for simplicity $r_n^H = \mathbb{I}[r^H[i] = n]$

$$\begin{aligned} \Phi(C) &= \sum_{m,n} W_{mn} \frac{C_{mn}}{E} \\ &= \frac{1}{E} \sum_{m,n} W_{mn} * C_{mn} \\ &= \frac{1}{E} \sum_{i \in \mathcal{T}} \sum_{m,n} W_{mn} \mathbb{I}[r^H[i] = m \wedge r^A[i] = n] \\ &= \frac{1}{E} \sum_{i \in \mathcal{T}} \sum_{m,n} W_{mn} \mathbb{I}[r^H[i] = m] \mathbb{I}[r^A[i] = n] \\ &= \frac{1}{E} \sum_{i \in \mathcal{T}} [-r_+^H r_-^A + -2r_+^H r_-^A - r_-^H r_-^A \\ &\quad + r_-^H r_+^A + 2r_-^H r_+^A + r_-^H r_-^A] \end{aligned} \quad (10)$$

Let us now expand the right side:

$$\begin{aligned}
\Phi(C) &= \frac{\text{mar}(\hat{\mathbf{d}}) - \text{mar}(\mathbf{d})}{E} \\
&= \frac{1}{E}((\hat{d}_+ - \hat{d}_-) - (d_+ - d_-)) \\
&= \frac{1}{E}(\sum_{i \in \mathcal{T}} r_+^A - \sum_{i \in \mathcal{T}} r_-^A - \sum_{i \in \mathcal{T}} r_+^H + \sum_{i \in \mathcal{T}} r_-^H) \\
&= \frac{1}{E} \sum_{i \in \mathcal{T}} (r_+^A - r_-^A - r_+^H + r_-^H)
\end{aligned} \tag{11}$$

It is now apparent that we need to find an equivalence of the terms inside the sum. That is:

$$\begin{aligned}
&-r_+^H r_-^A + -2r_+^H r_-^A - r_-^H r_-^A \\
&+ r_-^H r_+^A + 2r_-^H r_+^A + r_+^H r_+^A \\
&= r_+^A - r_-^A - r_+^H + r_-^H
\end{aligned} \tag{12}$$

We note that on the left hand side only at most one term of the sum can be non-zero. On the right hand side at most one term for the automated metric can be 1 at the same time, and one for the human rating. We complete the proof by simple enumeration of all possible values of r_n^A and r_m^H .

r_+^H	r_-^H	r_+^A	r_-^A	LHS	RHS
0	0	1	0	0	0
0	0	1	1	1	1
0	0	0	0	2	2
0	1	0	0	-1	-1
0	1	0	1	0	0
0	1	1	0	1	1
1	0	0	0	-2	-2
1	0	0	1	-1	-1
1	0	1	0	0	0

Thus, for all possible combinations of r_n^A and r_m^H , both sides yield the same value, thus, showing the equality of the two definitions.

B Proof of No System Level Sing Change

Here, we proof Lemma 2, i.e., that

$$\begin{aligned}
\Phi(C) &= 0 \\
\implies \mathbb{I}[\text{sgn}(d_+ - d_-), \text{sgn}(\hat{d}_+ - \hat{d}_-)] &= 1
\end{aligned} \tag{13}$$

The proof is a simple application of the alternative definition of the Favi-Score.

$$\begin{aligned}
\Phi(C) &= 0 \\
\iff \frac{\text{mar}(\hat{\mathbf{d}}) - \text{mar}(\mathbf{d})}{E} &= 0 \\
\iff \text{mar}(\hat{\mathbf{d}}) - \text{mar}(\mathbf{d}) &= 0 \\
\iff \text{mar}(\hat{\mathbf{d}}) - \text{mar}(\mathbf{d}) &= 0 \\
\iff \text{mar}(\hat{\mathbf{d}}) &= \text{mar}(\mathbf{d}) \\
\iff \hat{d}_+ - \hat{d}_- &= d_+ - d_- \\
\implies \text{sgn}(\hat{d}_+ - \hat{d}_-) &= \text{sgn}(d_+ - d_-)
\end{aligned} \tag{14}$$

Note that the last line of the proof is an implication, which is not reversible. Thus if the sign of both margins are equal, we cannot conclude that there is no favoritism.

C Experimental Setup Details

C.1 Convert Scalar to Preference

Since most metrics return a scalar value, we need to transform them into a preference rating. Similarly, in many cases human ratings are available as scalar or ordinal judgements for a single input and output pair. In these cases we assume that we have paired ratings, meaning that we have system output and rating for each test input and all systems. We can then compare the individual scalar (or ordinal) ratings to derive a preference rating, assuming that higher values indicate higher quality.

Definition 12 (Scalar Metric). *We call real valued functions of inputs and outputs **scalar metrics**:* $\mathcal{M}_s : \mathcal{I} \times \mathcal{O} \rightarrow \mathbb{R}$.

A preference metric can be constructed from a scalar metric as follows:

Definition 13. *The **derived preference metric** $\mathcal{M}$ of a given scalar metric $\mathcal{M}_s$ is defined as*

$$\mathcal{M}(i, o_1, o_2) = \begin{cases} + & \mathcal{M}_s(i, o_1) > \mathcal{M}_s(i, o_2) \\ = & \text{otherwise} \\ - & \mathcal{M}_s(i, o_1) < \mathcal{M}_s(i, o_2) \end{cases}$$

C.2 WMT-22 ChatGPT

Additionally to metrics participating in the WMT-22 metrics shared task (Freitag et al., 2022), we include ratings from GPT3.5-Turbo (0613) (OpenAI, 2023) prompted as a preference metric (Wang et al., 2024). We provided both a static *system prompt* describing the rating task and a *user prompt* for a concrete source sentence and pair of target translations. We used the following system prompt:

You act as an expert translator giving detailed feedback about candidate translations provided by users. Consider in particular the spelling, grammar, accuracy, and fluency of different translations. Make sure you give detailed information why one translation might be better than another.

We used the following user prompt, where the placeholders *source*, *hyp_a*, and *hyp_b* were replaced with a concrete source sentence and translations from systems π_1 and π_2 :

Please give feedback for the following translations:

Original Sentence: {source}

Candidate A: {hyp_a}

Candidate B: {hyp_b}

We used the *function calling* API (OpenAI, 2023) to get a structured output rating and asked the model to provide its preference (*Candidate A*, *Candidate B*, or *No Preference*), specific feedback for each candidate translation, and an explanation for its decision. For our evaluation, we discard the feedback and explanation texts which were included to elicit behavior analogous to *chain of thought* (Wei et al., 2022) prompting. We note that we received preference ratings that were uniquely interpretable as one of the 3 options in almost all cases. For 3 items, the model responded with an empty string as preference rating, which we assigned as *No Preference*.

For our experiments, we assume that the ratings are symmetric with respect to system order, meaning that if we have already collected the ratings for the pair (π_1, π_2) , we invert the ratings (+ to – and vice-versa) for the pair (π_2, π_1) . We enumerated systems in lexicographical order.

C.2.1 Symmetry of Preference Ratings

In the main text we treat the ratings from GPT3.5-Turbo as symmetric. This is mainly to keep the analysis simple. We note that according to Definition 13 the other preference metrics are already symmetric.

We nevertheless also collected ratings for flipped system pairs and noticed discrepancies. We show the confusion matrix in Figure 7. The overall accuracy is 69% and the intra-rater Krippendorff- α (Krippendorff, 1970) is 0.249.

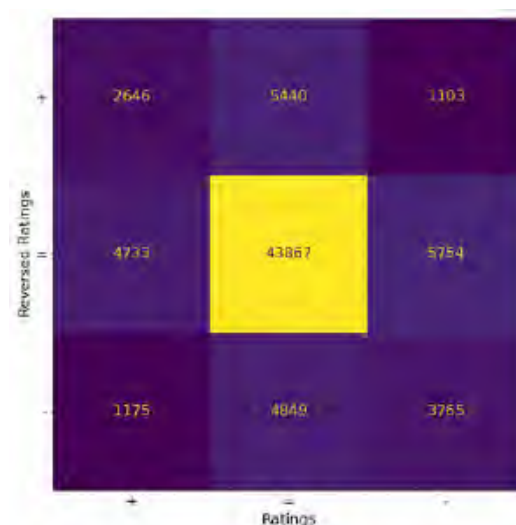


Figure 7: Confusion Matrix of reversed ratings with respect to the original GPT3.5 Turbo ratings.

D Examples of Favoritism Consequences

In this section, we show more examples of DAGs with the system-wise Favi-Score box plots.

D.1 Dialogue

The DAG according to human ratings is depicted in Figure 8a. The human reference is on the top row, while *DialoGPTLarge* is rated as the worst system according to the human ratings. According to the DEB Metric (Sai et al., 2020), human responses are rated much lower than according to the human raters, while the *PolyEncoder* is rated better than the *SeekerDial3B*. This matches the Favi-Scores depicted in Figure 8c, where the human reference is clearly disfavored, and the *PolyEncoder* has an overall positive Favi-Score. The *DialoGPTLarge* is also favored by the DEB metric, however, it does not have an impact on the new ranking. We also note that *PolyEncoder* is rated as being better than *SeekerDial3B* according to the metric, which is also a consequence of favoritism, as *SeekerDial3B* is heavily disfavored. In fact, the Favi-Score between the two systems lies at 0.72 in favor of *PolyEncoder*.

In Figure 9, the analysis for the MAUDE (Sinha et al., 2020) metric is depicted. MAUDE clearly prefers the outputs of *PolyEncoder*, which is also clear from the Favi-Score. Similar to DEB, MAUDE also prefers the outputs of *DialoGPTLarge*, however without changing the ranking. MAUDE has no clear disfavor against the human

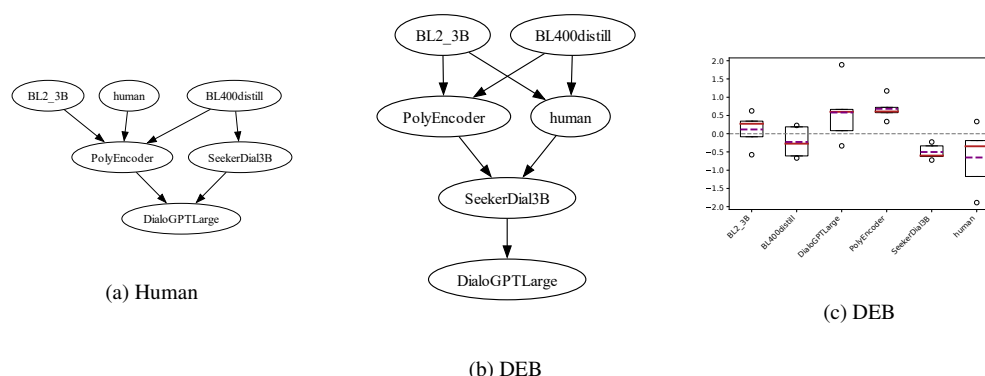


Figure 8: Visualization of the consequences of unfair metrics by comparing the human ranking to the DEB ranking. The ranking of a set of systems is depicted as a Directed Acyclic Graph, where an edge from system A to system B states that system A "wins" against system B. Here, a win is determined by a sign test (Coakley and Heise, 1996) at a 95% confidence threshold.

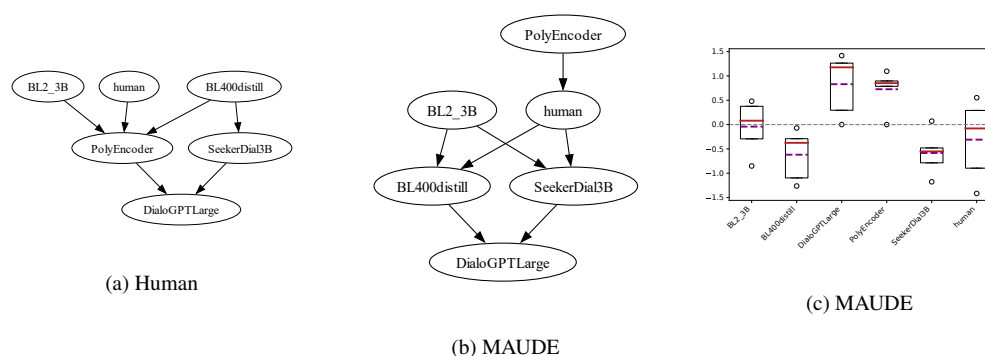


Figure 9: Visualization of the consequences of unfair metrics by comparing the human ranking to the MAUDE ranking. The ranking of a set of systems is depicted as a Directed Acyclic Graph, where an edge from system A to system B states that system A "wins" against system B. Here, a win is determined by a sign test (Coakley and Heise, 1996) at a 95% confidence threshold.

references like the DEB metric. Also *BL2_3B* is favored compared to *BL400distill* with a Favi-Score of 0.375 in favor.

D.2 Summarization

Figure 10 shows the analysis for BertScore (Zhang et al., 2019) on the SummEval data for the Consistency feature. The strongest favoritism is shown towards *M11* (a decoder focused on promoting novelty), which is the worst according to the human ranking, but is in solid mid-field according to the metric. While *M20* (a GPT2 model) is strongly disfavored by the metric. We also note that BertScore has a high system-level sign accuracy (0.789), while having an exceptionally low sample level accuracy (0.05). However, since the

Favi-Score is low as well (mean of 0.29), it showcases that the Favi-Score is more influential on the final ranking than the sample level accuracy.

Figure 11 shows the analysis of the ROUGE score. Overall, the ROUGE score has a slightly higher sample-level accuracy (0.07) while having a higher average Favi-Score (0.339), and a much lower system level sign accuracy (0.633). We also note that the favoritism is more pronounced in the ROUGE score, again with *M11* being strongly favored. While *M0* and *M1* are both disfavored, which also is apparent in the final ranking.

D.3 Machine Translation

In Figure 12 the analysis of ChatGPT as metric is shown. ChatGPT has a high system-level accuracy

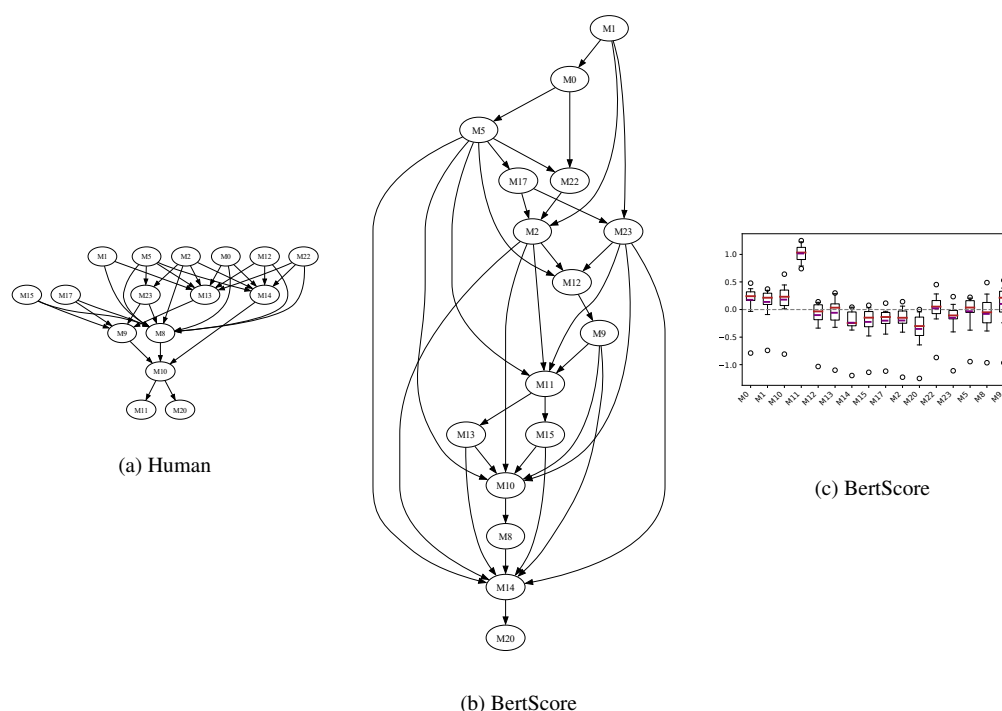


Figure 10: Visualization of the consequences of unfair metrics by comparing the human ranking to the BertScore ranking. The ranking of a set of systems is depicted as a Directed Acyclic Graph, where an edge from system A to system B states that system A "wins" against system B. Here, a win is determined by a sign test (Coakley and Heise, 1996) at a 95% confidence threshold.

(0.852), and a relatively low Favi-Score (0.229), as well as a low sample level accuracy (0.348). Among the selected WMT-22 metrics, it has one of the lower system-level accuracy scores, the highest sample-level accuracy score, and the highest Favi-Score. Thus, highlighting the influence of favoritism on the final ranking. Similar to other metrics, it disfavors the human reference (*ref-B*). It favors *M2M100*, however without chaining the ranking. These are cases where most mistakes are in favor of *M2M100*, however, the outcome margin is too large for the errors to have an impact on the ranking.

Figure 13 shows the analysis according to the BLEU score. The BLEU score has the lowest system level sign accuracy (0.7), and one of the higher Favi-Scores (0.21), and sample-level sign accuracy comparable to the other metrics (0.31). Again, we observe that the human reference is rated very poorly, with a high Favi-Score against it. Also *Online-W* has a negative Favi-Score, which also reflects its low position in the ranking according to BLEU.

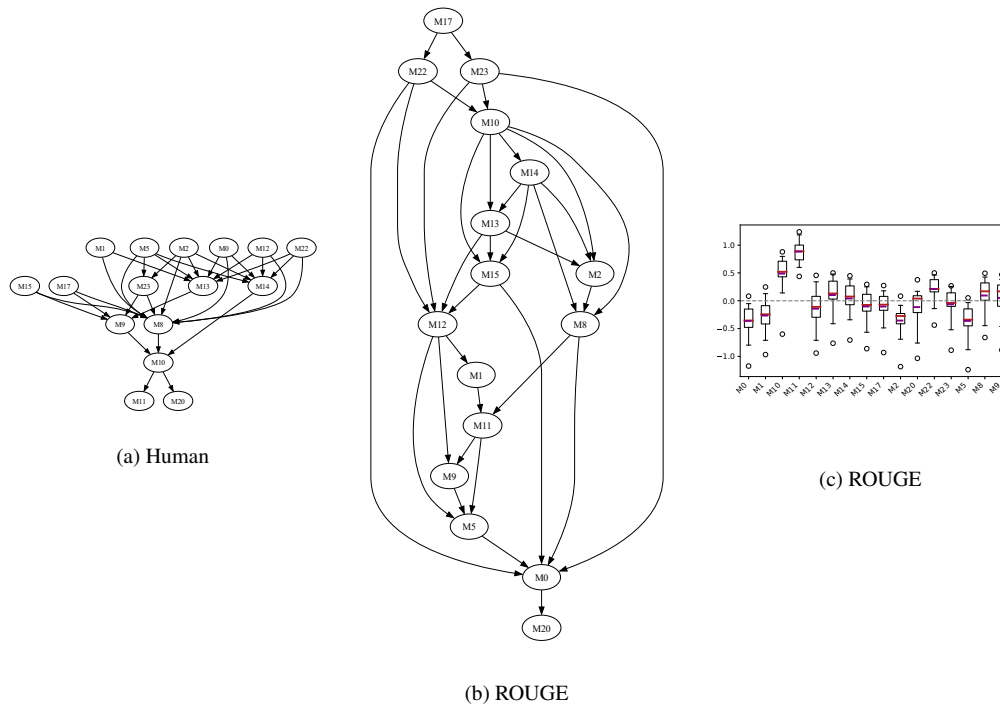


Figure 11: Visualization of the consequences of unfair metrics by comparing the human ranking to the ROUGE ranking. The ranking of a set of systems is depicted as a Directed Acyclic Graph, where an edge from system A to system B states that system A "wins" against system B. Here, a win is determined by a sign test (Coakley and Heise, 1996) at a 95% confidence threshold.

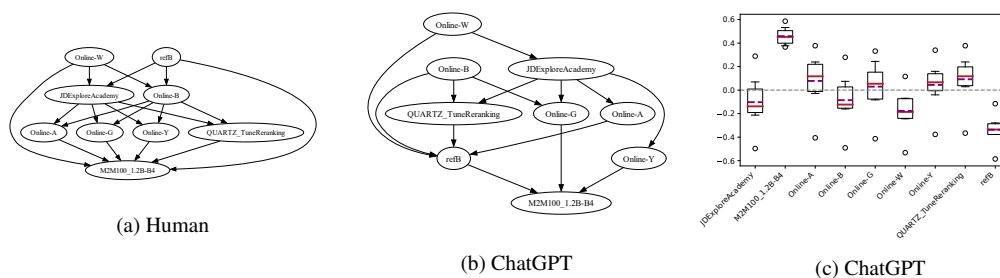


Figure 12: Visualization of the consequences of unfair metrics by comparing the human ranking to the ChatGPT ranking. The ranking of a set of systems is depicted as a Directed Acyclic Graph, where an edge from system A to system B states that system A "wins" against system B. Here, a win is determined by a sign test (Coakley and Heise, 1996) at a 95% confidence threshold.

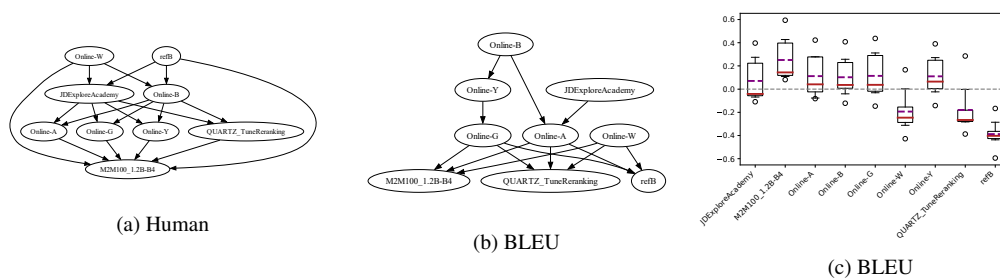


Figure 13: Visualization of the consequences of unfair metrics by comparing the human ranking to the BLEU ranking. The ranking of a set of systems is depicted as a Directed Acyclic Graph, where an edge from system A to system B states that system A "wins" against system B. Here, a win is determined by a sign test (Coakley and Heise, 1996) at a 95% confidence threshold.

Role-Playing LLMs in Professional Communication Training: The Case of Investigative Interviews with Children

Don Tuggener¹, Teresa Schneider², Ariana Huwiler³, Tobias Kreienbühl³,
Simon Hischier³, Pius von Däniken¹, Susanna Niehaus²

¹ Zurich University of Applied Sciences (ZHAW), Centre for Artificial Intelligence (CAI)

² Lucerne University of Applied Sciences and Arts (HSLU), Institute of Social Work and Law

³ Lucerne University of Applied Sciences and Arts (HSLU), Immersive Realities Research Lab

don.tuggener@zhaw.ch, susanna.niehaus@hslu.ch

Abstract

We present a novel approach for professional communication training in which Large Language Models (LLMs) are guided to dynamically adapt to inappropriate communication techniques by producing false information that match the biased expectations of an interviewer. We achieve this by dynamically altering the LLM's system prompt in conjunction with a classifier that detects undesirable communication behaviour. We develop this approach for training German speaking criminal investigators who interview children in alleged sexual abuse cases. We describe how our approach operationalises the strict communication requirements for such interviews and how it is integrated into a full, end-to-end learning environment that supports speech interaction with 3D virtual characters. We evaluate several aspects of this environment and report the positive results of an initial user study.

1 Introduction

Professional communication is subject to behaviour rules and linguistic registers (Holmes and Marra, 2014; Khranchenko, 2019; Bhatia and Bremner, 2012). Acquiring and training the skills to be proficient in professional communication can be a long, resource-intensive, and cumbersome road. Chatbots and virtual characters have emerged as a method to make professional training more accessible and cost-efficient in comparison to in-person training with human actors (Pompedda et al., 2022). One important factor for communication training is that the trainees can express themselves freely, i.e. using their own voice, words, and approach to a task rather than being presented with a selection of predetermined and fixed dialogue choices. In turn, it is important that the feedback on their performance is adapted and personalised to the individual conversational behaviour of the trainees. Consequently, virtual characters have to be able to dynamically re-



Figure 1: Screenshot of the training environment.

spond to different kinds of conversational behavior in a professional communication task.

In this paper, we explore the use of Large Language Models (LLMs) as the dialogue component in a sensitive professional communication situation, i.e. training criminal investigators in interviewing alleged child victims of sexual abuse. When children are interviewed about alleged experiences of sexual abuse, the quality of the investigative interview is crucial to whether their statements can be used as the basis for a criminal investigation (Korkman et al., 2024; Niehaus et al., 2017). This is because the child's statements are usually the only evidence in such proceedings (Steller, 2008). The demands on the quality of interviews and the qualifications of interviewers are correspondingly high.

Many training programs have been developed to improve interview quality in child interviews (Benson and Powell, 2015, e.g.). Elaborated and effective training programs include watching commentaries and videos of children being interviewed, quizzes, and mock interviews with colleagues or trained actors (Benson and Powell, 2015; Lamb, 2016). However, the latter is difficult to realise when it comes to training child interviews, as role-playing with fellow trainees is not realistic, and children cannot be used as actors for interviewer

training on the subject of abuse for ethical reasons. Investigators are currently forced to gain their initial experience on real cases, meaning that children allegedly affected by sexual abuse are often confronted with inexperienced interviewers (Niehaus et al., 2017). We therefore aimed to develop virtual characters with which optimal interviewing behaviour can be trained realistically and individually without risk before working on real cases. Through systematic and automated feedback from the system, investigators should learn to apply appropriate questioning techniques and avoid suggestive questions which may render the testimony useless as evidence and, in the worst case, stimulate the development of false memories. This training software is intended to contribute to an improvement in interviewing practice in order to meet the international demands on child-friendly justice (FRA, 2017).

2 Related Work

Three different training approaches have been developed to train interview behaviour in cases of suspected abuse with virtual characters that represent children. Pompedda et al. (2015) developed the “Empowering Interviewer Training” (EIT) in which the characters have predefined memories and responses that include relevant and neutral details. The characters answer using predefined response algorithms which are based on empirical knowledge about reactions to suggestive questioning. In the original version, a human operator needed to categorise the question that was asked by the participant. In a new version of the program, an automated question classification algorithm was tested (Haginoya et al., 2023). Overall, research found that the EIT combined with feedback increased the proportion of recommended questions and decreased the proportion of non-recommended questions asked by participants (Pompedda et al., 2022).

A similar system is also used in a more recent approach, an interactive virtual reality training called “ViContact” (Krause et al., 2024). However, as in the EIT, the responses remain limited to predefined memories and responses which are selected based on an algorithm after a human operator has categorised the question. New to the training is the 3D approach (i.e., virtual reality), that the interviewer needs to find out whether sexual abuse or another stressful event happened, and that participants are asked to build rapport with the child avatar before talking about the critical event. Although both pro-

grams have shown improvements in interviewing behaviour, the response generation is inflexible, the conversation flow is constrained through the prerecorded video sequences, and elaborated false memories cannot be produced. Furthermore, a human operator is usually needed to categorise the questions asked.

To tackle these problems, another research group is developing an AI-driven system that can dynamically handle questions, provides higher realism of the answer behaviour and does not need an operator (Hassan et al., 2022a). This approach utilises advanced natural language processing and provides an immersive experience through virtual reality. Several user studies cover the ongoing development of the child avatars (Hassan et al., 2022b; Salehi et al., 2022; Hassan et al., 2023; Røed et al., 2023; Salehi et al., 2024).

Although this newly developed AI-driven system can dynamically handle questions and provide feedback automatically without an operator, it only answers suggestive questions with a vague and unproductive reply. Like the EIT, it does not fabricate new false information when inappropriate questions are asked. This means that elaborated false memories¹ are not produced by the system.

In this paper, we introduce an AI-driven system that is based on a LLM, can dynamically answer questions based on the interview context and its knowledge, dynamically generates emotions based on the context and its own utterances, does not need an operator, and produces false memories when inappropriate questions are asked repeatedly. In summary, our contributions are as follows:

- We introduce the notion of generating *false memories* as a pedagogical tool in the training process. False memories occur when trainees apply inappropriate suggestive questioning and can lure trainees into drawing incorrect conclusions.
- We present a novel approach to steer LLMs through *altering the system prompt dynamically* in conjunction with a classifier that detects inappropriate conversational behaviour.
- We outline and implement a practical approach for the *efficient selection of an LLM* based on technical and qualitative requirements for our setting.

¹In the following, the term false memories is not used in the forensic sense of a pseudo-memory. In the context of our study, we refer to the reactive (forensically more comparable to compliance) production of partially or completely false information that can alter memories in the long term.

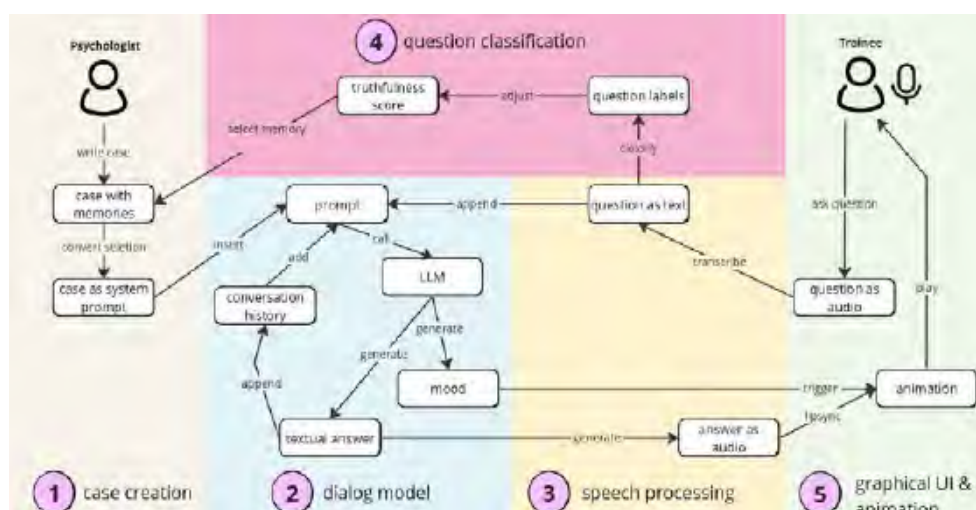


Figure 2: Overview of the architecture of the training system and its components. The case, created by forensic psychologists (1) is entered into the dialog model (2). Questions (by trainee) or answers (generated by the dialog model) pass through speech processing (3) where they are turned into audio or text respectively. Question classification monitors the trainee's utterances for inappropriate content and adjusts the truthfulness score (4). This score is used to trigger the injection of false memories from the case description into the LLM prompt, if inappropriate content is detected. Answers and moods generated by the LLM are passed into the graphical UI & animation component, where the character is animated and shown to the user (5).

- Finally, we release a *dataset for problematic utterance classification* in children interviews in German.²

3 Approach

Our setting can be seen as conversational information retrieval, i.e. the user wants to elicit information about a specific case from the system. However, our system is reluctant to provide the information and needs to be prompted in a certain way. Failure to do so inflicts the system's willingness to cooperate. In fact, inappropriate questioning yields false information that misleads the user into drawing incorrect conclusions about the case at hand, while open prompts for narration increase the chance of uncovering the facts of the case. This comprises the overall pedagogic intent of our system: interviewers should learn how to question children in an appropriate way without distorting the statements.

3.1 True Memories

Our virtual character's memory is structured into a semantic memory and episodic memory. The semantic memory contains static information about

its situation regarding family, hobbies etc. This information is verbalised in an unordered set of utterances in the first person perspective of the character, e.g. "I am 4 years old", "I like playing tennis.", etc. The utterances can be used to answer a set of similar or related questions. For example, the utterance "I am 4 years old" can be retrieved to answer differently phrased questions about the character's age³, etc. However, the goal of writing these statements is not to anticipate all potential questions, but to outline a personality on the basis of which a dialog model will be tasked to role-play a character. The pedagogic purpose of the semantic memory is to enable the interviewer to establish rapport with the child, which is a crucial step in the initial phase of the interview.

The episodic memory contains information about the sequence of the event that is the topic of the interview, i.e. information about the alleged sexual abuse and its context. This information is also saved in the form of first person utterances, such as "I went to the basement with my teacher."

3.2 False Memories and Truthfulness

One central aspect of this design is that it allows for the incorporation of false memories. If an in-

²Available at <https://drive.switch.ch/index.php/s/DCMIo3SnnNcksQi>

³As we will see later, this statement can also be used to infer whether the character goes to school, etc.

interviewer applies inappropriate questioning techniques repeatedly, the character will start to confirm explicit suspicions of abuse although they are not confirmed in the original storyline. To this end, each episodic memory is accompanied by a false memory storyline that can be triggered by inappropriate questioning style.

At the core of the virtual character's behaviour is the truthfulness score. It determines whether the character is answering truthfully or gives false information. The truthfulness score is adjusted according to the interviewer's questions in a penalty/reward system. At the beginning of the interview, the character is in a neutral and truthful state. If questioned appropriately, it returns truthful and factual answers. If problematic and inappropriate questions are detected, the score is lowered, depending on the severity of the suggestive content of the question: Mentioning the suspect and sexual abuse in a question before the character reveals such information yields the highest score deduction, while asking about a specific point in time or posing a forced-choice question only minimally decreases the truthfulness score. If the score drops below a preset threshold, the character starts generating unreliable responses.

For example, when questioning a 4 year-old virtual character in whose case no abuse occurred, the truthfulness score starts at 10. A suggestive question with a sexual keyphrase that was not uttered by the character itself beforehand, such as "Did you have to take off your pants?" (take off + cloth), will reduce the score by 3 points. This is already under the preset threshold of 8 and the virtual character will start to include incorrect details in their answers. If three unproblematic questions are asked subsequently, the score rises and the character will again respond truthfully. If more inappropriate questions are asked and the score drops below 4, the character's truthfulness cannot be restored and its reported story remains distorted.

4 Implementation

In the following sections, we outline the technical implementation of the approach outlined above. Figure 2 shows an overview of the system⁴.

4.1 Dialog Model

The dialog model encompasses the following tasks:

- Managing the character's memory
- Detecting inappropriate and appropriate questioning
- Generating answers to questions in accordance with the two points above
- Generating appropriate emotions tags that steer the 3D animation of the virtual character

While it seems tempting to implement all functionality in one "mega" prompt for LLMs given their ever increasing capabilities, early experiments quickly revealed, in line with Khot et al. (2022), that such a highly complex set of tasks needs to be decomposed. Below, we outline the different components of the dialog model and how they interact.

4.1.1 Character Memory

A key differentiator from related work in our approach is that our characters dynamically respond to inappropriate questioning by yielding false information that confirms biased suspicions of the interviewer. For each case, in addition to the truthful version of the story, the forensic psychologist write two other storylines, depending on whether the case contains abuse:

Cases without abuse contain a truthful storyline without abuse and two additional ones, where the 1st alternation contains comparably less severe forms of abuse and the 2nd version confirms explicit and severe sexual abuse.

Cases with abuse initially contain a storyline that does not explicitly state the abuse, but hints at it. The interviewer first has to establish trust and rapport with the character (by asking appropriate questions such as narration prompts) to unlock the truthful storyline that contains the abuse. As in the cases without abuse, inappropriate questions alter the story. The 1st alternation contains ambiguous hints and the 2nd version contains more severe abuse than the truthful one and the 1st alternation.

4.1.2 Dialog Model

Anticipating and writing out questions that might be asked by interviewers and all the potentially ensuing dialog branches in the different storylines is infeasible; especially given the fact that several cases are needed for training purposes. Hence, implementing the dialog model with an approach where the questions posed are matched to preset questions to retrieve an answer (Bosse and Gerritsen, 2017; Barbe et al., 2023) is impractical. Also, preparing the storylines in such a way that all statements can be retrieved individually independent of

⁴See Appendix A.1 for a technical description of the 3D characters.

the context leads to utterances sounding unnatural⁵.

Fortunately, the advent of Large Language Models (LLMs) like ChatGPT⁶ gave rise to dialog models that are pre-trained on large amounts of human conversations and can thus handle their intricacies gracefully. We leverage LLMs by ingesting a character's semantic and episodic memories into the LLM via the *system prompt*. We developed a system prompt⁷ that contains the semantic and episodic memories of a character, as well as instructional behaviour.

However, including the semantic memory and all story variants of a character in the system prompt grows it to an unmanageable size and places a large burden on the LLM to manage it. Hence, we developed a mechanism to adapt the system prompt in accordance with the behaviour of the interviewer. Specifically, the system prompt contains a placeholder variable for the episodic memory. At the start of the conversation, this variable is filled with the truthful story of the character and the character's truthfulness score is set to default. If the score drops below a preset threshold during the interview, the placeholder variable for the episodic memory is filled with an alternate storyline, that is, the memory of the character begins to change and it provides false information. However, the conversation history between the interviewer and the virtual characters remains intact.

4.1.3 LLM Selection

Our goal is to find an LLM that suits our needs without having to perform vast amounts of experiments and manual annotations. Hence, rather than creating large benchmarks, we define the minimal set of technical and qualitative requirements and design specific probes for them. After discussing various forensic and technical aspects, we defined the following technical requirements:

Convenience: SDK support, low-latency APIs, affordable pricing, generous rate limits.

Context window size: Providing a large enough context window to fit the rather long system prompt and the rather long following conversation.⁸

⁵In preparing statements for matching approaches, it is not permissible to write utterances like: "Mr. Smith is my teacher. I like him a lot." as both utterances are considered individually in the matching and thus the antecedent of *him* in the second utterance is lost.

⁶<https://openai.com/blog/chatgpt>

⁷See Appendix A.2.

⁸The context window size is the maximum number of words that can be sent as a request for an answer to an LLM

Language support: Support the German language.

Alignment: Ability to discuss sensitive topics (sexual abuse).

In addition, we identified the following qualitative abilities:

Natural use of language: Understanding and using *deixis* (i.e. referential expressions like pronouns). *Adapting the pre-set statements* of the system prompt to the conversational context rather than citing a declarative sentence from the memory verbatim. Speaking in *age-appropriate language* regarding the preset age of the character.

Role-Playing: Staying in character and following the instructed behaviour (e.g. not outputting any meta-commentary or reference to the training setting, etc.). *Handle unforeseen questions* that tackle information that is not part of the predefined semantic memory gracefully (e.g., "Where do you live?")

Factuality: *Adhering to the given memories* (semantic and episodic), i.e. *avoiding hallucinations* that contradict the given memories (while being allowed or even encouraged to answer unforeseen questions).

Long conversations: Holding natural, consistent *long multi-turn conversations*⁹

4.1.4 Technical Requirements

The **Convenience** requirement narrowed our selection to the following providers (and models): Google (Chat Bison/Gemini)¹⁰, OpenAI (ChatGPT/GPT-4)¹¹, Anthropic (Claude 3)¹², Mistral (Mistral/Mixtral)¹³, and Meta (Llama 2)¹⁴. In initial tests, we noticed that Claude 3's bigger models (Sonata and Opus) have quite strict rate limits given our usage tier. We therefore settled on the smallest model in the family, Claude 3 Haiku. Regarding GPT-4, we noticed that the latency was quite high at times and the pricing seemed prohibitive. Also, we did not observe stark quality differences to ChatGPT 3.5 in our initial tests. Therefore, we chose ChatGPT 3.5 as the candidate. Fi-

and contains the system prompt, the conversation history, and the current user statement that needs answering.

⁹The degradation of answers in longer conversations is a known problem of many machine learning-based dialogue systems. (Spataru et al., 2024).

¹⁰<https://cloud.google.com/vertex-ai>

¹¹<https://platform.openai.com/>

¹²<https://console.anthropic.com>

¹³<https://console.mistral.ai/>

¹⁴<https://llama.meta.com/llama2/>

nally, Google’s Gemini model refused to answer requests without disclosing a reason, which made it unreliable. Therefore, we settled on Chat Bison.

We found that models have a sufficient **Context window size**¹⁵ and that they support **German**. Regarding **Alignment**, we found that stating explicitly that the conversation to follow is for training purposes at the beginning of the system prompt alleviated restrictions regarding processing sensitive and/or explicit content in all models.

4.1.5 Qualitative Requirements

Since evaluating the remaining requirements quantitatively would require resources beyond the scope of our project, we explore them in a qualitative and comparative manner. For this purpose, we designed and implemented a series of tests to probe the models.

To get an initial impression of the models’ capabilities, we leveraged the Eden AI platform¹⁶ which provides an easy to use interface¹⁷ to elicit answers from various LLMs. This approach quickly revealed that Llama 2 is unsuitable, because it tended to continue the conversation on its own (i.e. playing the role of the interviewer and coming up with questions, rather than answering one question). Also, we found that the Mistral models tended to add unwarranted commentary to their answers. Thus, we eliminated these two models from the set of candidates. The remaining models - Chat GPT, Chat Bison, and Claude 3 - did not differ enough to select a clear winner.

Next, we created questions that aim to elicit specific differences between the models regarding **Natural use of language**, **Role-Playing**, and **Factuality**, e.g. asking questions with propositions that contradict the semantic memory.¹⁸ We then did a comparative ranking (Li et al., 2019; Chiang et al., 2024, e.g.) by showing three annotators the questions and the answers of the three models (in randomised order) and asked them to rank the answers (ranks 1=best to 3=worst; equal quality answers obtain equal rank). We then calculated the average ranks of the models’ answers across all annotators. Figure 3 shows the results.

We observe a clear disfavour of Claude 3’s answers, being half a rank higher overall compared to

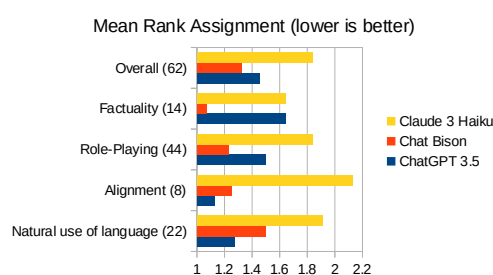


Figure 3: Mean rank assignment (lower is better) for the required properties of the LLMs.

the other two models. ChatGPT and Chat Bison are ranked similarly overall. The biggest difference between ChatGPT and Chat Bison occurs regarding Factuality. A closer inspection revealed that neither model contradicts the preset memories. However, they answer differently. The question “How was school today?” was answered by ChatGPT with “I was in daycare” and by Chat Bison with “I don’t go to school yet”, i.e. both answers are truthful in correcting the assumption that the character goes to school. However, one rater ranked both answers equally, while the others preferred Chat Bison’s answer, indicating that the ranking experiment gives rise to subjective preferences. Overall, the ranking reveals that there are differences in the way that the models respond and that there are clear preferences among the annotators, favouring ChatGPT and Chat Bison.

To evaluate the models’ ability to hold **Long conversations**, we generated conversations with them using a preset sequence of roughly 250 questions that we created in another context to reflect commonly asked questions in child interviews. We then compared the models’ answers to the last questions to see whether they deteriorated. Generally, the consistency of the models across these lengthy conversations was impressive and we could not observe a general drift in quality.¹⁹

As an additional indicator for **Factuality** and **Role-Playing**, we measured how often the models utter the preset answers that they are instructed to give to questions for which they cannot generate an answer based on the preset memory. The models are instructed to answer such questions with

¹⁵See Appendix A.5 for how we calculated the required size.

¹⁶<https://app.edenai.run/bricks/text/chat>

¹⁷See Appendix A.6.

¹⁸E.g. asking “How was school today?” when the character is supposed to be 4 years old.

¹⁹We observed that Chat Bison sometimes started breaking character by saying, sometimes in English, that as a language model, it cannot judge certain propositions (e.g. “Is Minecraft a violent game?”, “Can the user kill others in Minecraft?”). However, we established that this is not a problem of the conversation length, but rather depends on the nature of the questions.

“What?” and “I don’t know.” or invent an ad hoc answer. We found that ChatGPT gave 49 “What?/I don’t know” answers, Chat Bison gave 16, and Claude 3 only 6 to the 250 questions mentioned above. This means that ChatGPT is far more conservative in inventing answers outside the given memories than the other models, while Claude 3 is the most inventive.

Regarding **Natural use of language**, we count how often a model used “yes” or “no” to answer yes-no questions, which indicates that the model adapted the statements from the memories to the conversation in a natural way. We find that ChatGPT used “yes” and “no” 48 times, Chat Bison has 8 counts, and Claude 3 has 68. That is, Chat Bison seems to struggle to infer how to use yes/no-answers.

Finally, we approximate how well the models uttered **age-appropriate answers in role-playing** by measuring the readability scores and stylistic properties of their answers (Schuster et al., 2020).²⁰ We assume that the better the readability score, the more likely it is that a model uses age appropriate language. We applied a tool²¹ that calculates readability and various stylistic features to the models’ answers to the above-mentioned 250 questions. Table 1 shows the results.

	ChatGPT	Chat Bison	Claude 3
ARI	3.19	3.44	6.49
words per sent.	6.17	6.24	10.83
type-token ratio	0.13	0.18	0.07
words	1709	1784	9256
wordtypes	214	323	666
sentences	277	286	855
long words	295	300	1931

Table 1: Automated Readability Index (ARI) scores and stylistic features of the LLMs’ answers.

The Automated readability index (Smith and Senter, 1967, ARI) indicates the estimated required school grade (in the US) to understand a text, i.e. a lower score means an easier text. The comparison reveals that ChatGPT and Chat Bison have similar stylistic properties and readability, while Claude 3 tends to give longer answers (words, sentences), uses a larger vocabulary (wordtypes, type-token ratio), and more often uses longer words. Based

²⁰Readability scores assess how easy or difficult texts are to read and take into account statistical features of texts, such as words per sentence, syllables per word, and use of punctuation etc.

²¹<https://github.com/andreasvc/readability>

on this analysis, Claude 3 seems less likely to give realistic age-appropriate answers than the other two models.

Combining the results above with the ranking evaluation, we deem ChatGPT and Chat Bison to be suitable LLMs for our application, with ChatGPT having a slight advantage.

4.2 Question Classification

Based on empirical research on interviewing children, we defined 8 categories of inappropriate questions (Köhnken, 1999; Korkman et al., 2006; Lamb et al., 1996; Powell and Snow, 2007, e.g.): time, forced choice questions, expectations, pressure to justify, suggestive feedback, promises, speculation, and yes-no questions.²² In addition, and similar to Haginoya et al. (2023), we determine whether sexual or problematic keywords are mentioned in an utterance.

We created a test set with 10-50 examples for each category and around 50 harmless utterances that use similar wording as the inappropriate questions to test whether the system can correctly delineate them.²³ In total, we created 200 utterances. Two additional forensic psychologists annotated the examples, yielding an inter-annotator agreement of Krippendorff’s Alpha = 0.74. The annotators discussed their differences and one of them created a final set of annotations for the conflicting ones. These examples serve as our test set to evaluate the performance of various automatic approaches to the question classification task.

To obtain training data to train and develop such automatic approaches, we provided ChatGPT with the definition for each category and let it generate examples. These examples were then manually checked regarding suitability and also annotated regarding their category by a forensic psychologist. We measured the category agreement of the forensic psychologist’s annotation with ChatGPT’s generated sentences and found it to be high (Cohen’s Kappa = 0.79). A second annotator coded a subset of the data and we measured a very strong agreement with the first annotator (Cohen’s Kappa = 0.92), deeming it a valid training set. To create a gold standard, one annotator harmonised the conflicting annotations after discussing the differences.

²²See Appendix A.3 Table 8 for the definitions and examples.

²³See Table 7 in the Appendix for detailed dataset statistics.

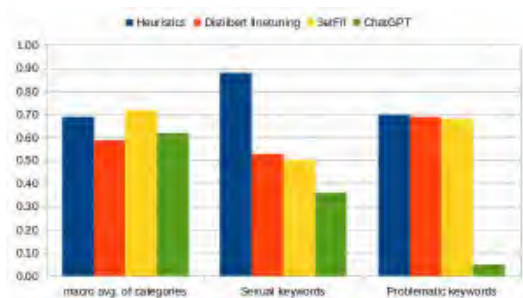


Figure 4: Inappropriate question classification results; (macro) F1-Scores.

4.2.1 Methods & Evaluation

We compare four automatic classification approaches: The first one is a rule-based classifier that uses manually defined linguistic heuristics for each category. This approach leverages a syntactic parser²⁴, lexical resources such as word lists (Klenner et al., 2009), and a textual similarity module (Reimers and Gurevych, 2019) to compare utterances to predefined examples. The second approach applies ChatGPT as the classifier. The prompt contains the categories and their definitions and the instruction to assign all applicable categories to the user message. Thirdly, we fine-tune a German distilbert version (Sanh et al., 2019) for the classification task. Finally, we use the distilbert model as the base to train SetFit (Tunstall et al., 2022), which is a classification model that works well for settings where little data is available.

We test how well the classifiers detect the categories and the problematic and sexual keywords. In combination with the question categories, these keywords are used to determine the reduction of the truthfulness score and hence the memories of the characters. Figure 4 shows the results for the test set.²⁵ For the categories, SetFit achieves the highest macro F1 score (0.72). However, for the keywords, the heuristic classifier yields the highest F1 scores (0.88 and 0.70). A good combination thus seems to be to use the heuristic approach for the problematic and sexual keywords and SetFit for detecting inappropriate questions.

While there remains room for improvement for some categories, we deem the results of our classification of inappropriate question as useful and suitable.

²⁴<https://spacy.io/>

²⁵For full details of the results, see Table 6 in Appendix A.3.

5 User Study

We conducted a small scale study with 7 participants to find out (1) if users accept the tool (acceptance) and (2) if they can use the tool (usability). The participants interacted with the system for 15 minutes and then filled in two questionnaires: (1) the system usability scale (Brooke, 1996, SUS)²⁶ with 10 questions and (2) a questionnaire on the acceptance of the technology with 6 questions. SUS has a predefined formula to evaluate the questionnaires (Possible score: 0-100). Our evaluation obtained a score of 76.07 (AVG), i.e. the score is in the upper 0.25 percentile (0.5 percentile equals a score of ~68), meaning the application performed better than 75% of other systems evaluated with SUS.²⁷

To evaluate acceptance, we relied on the questionnaire surrounding the Unified Theory of Acceptance and Use of Technology (Venkatesh et al., 2016, UTAUT). Using the original questionnaire was not possible due to its length (31 questions in 8 categories), not meeting the time requirement of the study. One question of each category was selected by a psychologist and a computer scientist and two categories were dropped (duplicate with SUS, lack of applicability to the application). Lastly, the questions were translated into German.²⁸

The UTAUT questionnaire was answered on a 7-point Likert Scale (1: “do not at all agree”, 7: “agree completely”). Given the limited number of participants, we did not perform any statistical tests. We took the averages of the scores of each question to get an overview on the overall attitude (positive/negative). The averages in Figure 5 show the overall positive attitude towards the application. Notably, participants reported high willingness (6.29) to use the tool independently for skill enhancement. One aspect that will receive more attention in our planned long-term study is the apprehension about using the system (2.43).

6 Conclusion

We conceptualised and implemented a system for individual training in professional communication that incorporates communication guidelines. We employed a Large Language Model (LLM) as the

²⁶The SUS is a well-established questionnaire to measure the usability of a system in a quick, low-cost manner, resulting in a score that indicates whether or not a system promotes usability.

²⁷<https://measuringu.com/sus/>

²⁸See Appendix A.4.

On a Likert Scale: 1 do not at all agree 7 completely agree	Using the tool allows me to learn doing child interrogations more competently	I liked using the tool	I think my employer would enable and support the use of this tool	The tool is compatible with other training materials on the subject of child interrogation	I feel apprehensive about using the system	If I could use the tool for independent skill enhancement during working hours I would do so
Average (n=7)	5.29	5.14	5.57	5.43	2.43	6.29

Figure 5: UTAUT results.

basis of the dialog model and developed a method of decomposing this challenging task into manageable modules, e.g. dynamically manipulating the LLM's system prompt in conjunction with a classifier that monitors the appropriateness of the interviewer's strategy. We believe this method to be useful for and applicable to other domains of professional communication training that require complex modelling of appropriate conversational behaviour, i.e. health care, job interviews, or counselling sessions, etc.

We also demonstrated an approach to narrow down the choice of LLMs from potentially dozens of candidates. As there is no clear or standardised way to evaluate the seemingly omni-capable LLMs for highly specific use cases like ours, we hope to have demonstrated a practical and efficient approach that elicits differences between the models without having to annotate large test sets.

Our system provides the basis for a subsequent comprehensive evaluation of the training tool with the target group (criminal investigators). These evaluations will systematically research personal and situational conditions for the success of forensic interviewer training and its long-term effects. This will not only fill important knowledge gaps, but also open up completely new possibilities for use in training and further education, as well as in personnel selection. Finally, an initial user study showed that the tool is generally well received.

Ethical Considerations

Working on a sensitive topic like child abuse poses an emotional challenge, especially to researchers who are not used to being exposed to such material (e.g. software developers, computational linguists). Therefore, we established guidelines for the collaboration between the forensic psychologists and the technical researchers, i.e. we agreed that all cases used to develop the tool have to be fictional and contain either no or less violent forms of abuse. Exposure to transcripts of real interviews where abuse

is reported was limited to a necessary minimum.

For the user study, we focused the experiment on the semantic memory of the virtual character and did not include an episodic memory that contains explicit sexual abuse. Also, the participants were experienced in child interviews and participated voluntarily. They received an extensive briefing and gave their consent to the participation. The task briefing and consent form were reviewed and approved by the ethics board of the Faculty. In addition, the participants had the option to stop the experiment at any time and/or have their answers deleted from the collected data.

Limitations

- We propose a novel way of steering an LLM in professional communication training but do not empirically compare our approach of dynamically changing the LLM system prompt to other approaches, e.g. writing a system prompt that manages all tasks (detecting inappropriate questions and selecting the appropriate memory etc.). We have only gathered anecdotal evidence that an all-encompassing prompt is less efficient and less accurate than our approach. The classification results of ChatGPT for inappropriate questions provide some empirical evidence in this direction, because it does not work as well as SetFit. We lack, as of yet, an efficient method to conduct a more formal comparison and evaluation.
- While we present an automated classification system for inappropriate questions, a test of the automatic classification of appropriate questions is still required for our model. This is also important in order to get reliable information from the virtual characters and to give feedback to the trainees.
- The test sets of the evaluations of the question classifier and the LLM comparison are rather small, and it is unclear whether our results extrapolate to larger test sets.
- It is, as of yet, unclear how to specifically evalu-

ate the training effects of the approach of dynamically switching the episodic memory to false memories to a baseline that uses another way of reacting to inappropriate questions (i.e. simply refusing to answer them). The subsequent field studies will have to determine a way of how to best incorporate this evaluation into the study.

- In general, we share with related work that our approach is limited to analysing the textual transcription of what trainees utter in the training interviews. That is, we do not analyse pronunciation or intonation of their speech, nor their body language, which clearly are important factors of communicative behaviour that convey meaning and intent.
- Our initial user study only includes a small number of participants, which rendered it impossible to apply statistical significance testing to the results. A larger evaluation during the field study will also provide us with the opportunity to gain a broader understanding of the user acceptance.

Acknowledgments

This work was funded by the Swiss National Science Foundation (SNSF) (SNF-Projektförderung / Projekt Nr. 189236) for the project "Virtual Kids - Virtual characters to improve the quality of child interviewing".²⁹

References

- Hermann Barbe, Jürgen L. Müller, Bruno Siegel, and Peter Fromberger. 2023. [An open source virtual reality training framework for the criminal justice system](#). *Criminal Justice and Behavior*, 50(2):294–303.
- Mairi S Benson and Martine B Powell. 2015. [Evaluation of a comprehensive interactive training system for investigative interviewers of children](#). *Psychology, Public Policy, and Law*, 21(3):309–322.
- Vijay K Bhatia and Stephen Bremner. 2012. [English for business communication](#). *Language Teaching*, 45(4):410–445.
- Tibor Bosse and Charlotte Gerritsen. 2017. Towards serious gaming for communication training—a pilot study with police academy students. In *Intelligent Technologies for Interactive Entertainment: 8th International Conference, INTETAIN 2016, Utrecht, The Netherlands, June 28–30, 2016, Revised Selected Papers*, pages 13–22. Springer International Publishing.
- John Brooke. 1996. SUS: A 'quick' and 'dirty' usability scale. In Patrick W. Jordan, Bruce Thomas, Bernard A. Weerdmeester, and Ian Lyall McClelland, editors, *Usability Evaluation in Industry*, chapter 21, pages 189–194. Taylor and Francis.
- Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E Gonzalez, et al. 2024. [Chatbot arena: An open platform for evaluating llms by human preference](#). *arXiv preprint*.
- FRA. 2017. *Child-friendly justice – Perspectives and experiences of children involved in judicial proceedings as victims, witnesses or parties in nine EU Member States*. Publications Office, Luxembourg.
- Shumpei Haginoya, Tatsuro Ibe, Shota Yamamoto, Naruyo Yoshimoto, Hazuki Mizushi, and Pekka Santtila. 2023. [Ai avatar tells you what happened: The first test of using ai-operated children in simulated interviews to train investigative interviewers](#). *Frontiers in Psychology*, 14.
- Syed Zohaib Hassan, Saeed Shafiee Sabet, Michael Alexander Riegler, Gunn Astrid Baugerud, Hayley Ko, Pegah Salehi, Ragnhild Klingenberg Røed, Miriam Johnson, and Pål Halvorsen. 2023. [Enhancing investigative interview training using a child avatar system: a comparative study of interactive environments](#). *Scientific Reports*, 13(1):20403.
- Syed Zohaib Hassan, Pegah Salehi, Michael Alexander Riegler, Miriam Sinkerd Johnson, Gunn Astrid Baugerud, PÅL Halvorsen, and Saeed Shafiee Sabet. 2022a. [A virtual reality talking avatar for investigative interviews of maltreat children](#). In *Proceedings of the 19th International Conference on Content-Based Multimedia Indexing*, pages 201–204.
- Syed Zohaib Hassan, Pegah Salehi, Ragnhild Klingenberg Røed, Pål Halvorsen, Gunn Astrid Baugerud, Miriam Sinkerd Johnson, Pierre Lison, Michael Riegler, Michael E Lamb, and Carsten Griwodz. 2022b. [Towards an ai-driven talking avatar in virtual reality for investigative interviews of children](#). In *Proceedings of the 2nd Workshop on Games Systems*, pages 9–15.
- Janet Holmes and Meredith Marra. 2014. The complexities of communication in professional workplaces. In *The Routledge Handbook of Language and Professional Communication*, pages 112–128. Routledge.
- Tushar Khot, Harsh Trivedi, Matthew Finlayson, Yao Fu, Kyle Richardson, Peter Clark, and Ashish Sabharwal. 2022. Decomposed prompting: A modular approach for solving complex tasks. In *The Eleventh International Conference on Learning Representations*.
- Dmitry S Khrumchenko. 2019. [Functional-linguistic parameters of english professional discourse](#). *Professional Discourse & Communication*, 1(1):9–20.

²⁹<https://www.hslu.ch/en/lucerne-university-of-applied-sciences-and-arts/research/projects/detail/?pid=5467>, <https://www.zhaw.ch/en/research/research-database/project-detailview/projektid/4072/>

- Manfred Klenner, Angela Fahrni, and Stefanos Petrakis. 2009. [PolArt: A robust tool for sentiment analysis](#). In *Proceedings of the 17th Nordic Conference of Computational Linguistics (NODALIDA 2009)*, pages 235–238, Odense, Denmark. Northern European Association for Language Technology (NEALT).
- Günter Köhnken. 1999. Suggestion und Suggestibilität. *Forensische Psychiatrie und Psychologie des Kindes- und Jugendalters*, pages 342–353.
- Julia Korkman, Henry Otgaar, Linda M Geven, Ray Bull, Mireille Cyr, Irit Hershkowitz, J-M Mäkelä, Michelle Mattison, Rebecca Milne, Pekka Santtila, et al. 2024. [White paper on forensic child interviewing: research-based recommendations by the european association of psychology and law](#). *Psychology, Crime & Law*. Advance online publication.
- Julia Korkman, Pekka Santtila, and N Kenneth Sandnabba. 2006. Dynamics of verbal interaction between interviewer and child in interviews with alleged victims of child sexual abuse. *Scandinavian journal of psychology*, 47(2):109–119.
- Niels Krause, Elsa Gewehr, Hermann Barbe, Marie Merschhemke, Frieda Mensing, Bruno Siegel, Jürgen L Müller, Renate Volbert, Peter Fromberger, and Anett Tamm. 2024. [How to prepare for conversations with children about suspicions of sexual abuse? evaluation of an interactive virtual reality training for student teachers](#). *Child Abuse & Neglect*, 149:106677.
- Michael E Lamb. 2016. [Difficulties translating research on forensic interview practices to practitioners: Finding water, leading horses, but can we get them to drink?](#) *American Psychologist*, 71(8):710–718.
- Michael E Lamb, Irit Hershkowitz, Kathleen J Sternberg, Barbara Boat, and Mark D Everson. 1996. Investigative interviews of alleged sexual abuse victims with and without anatomical dolls. *Child Abuse & Neglect*, 20(12):1251–1259.
- Margaret Li, Jason Weston, and Stephen Roller. 2019. [Acute-eval: Improved dialogue evaluation with optimized questions and multi-turn comparisons](#). *arXiv preprint*.
- Masahiro Mori, Karl MacDorman, and Norri Kageki. 2012. [The uncanny valley \[from the field\]](#). *IEEE Robotics & Automation Magazine*, 19(2):98–100.
- Susanna Niehaus, Renate Volbert, and Jörg M Fegert. 2017. *Entwicklungsgerechte Befragung von Kindern in Strafverfahren*. Springer, Heidelberg.
- Francesco Pompedda, Angelo Zappalà, and Pekka Santtila. 2015. [Simulations of child sexual abuse interviews using avatars paired with feedback improves interview quality](#). *Psychology, Crime & Law*, 21(1):28–52.
- Francesco Pompedda, Yikang Zhang, Shumpei Haginoya, and Pekka Santtila. 2022. [A mega-analysis of the effects of feedback on the quality of simulated child sexual abuse interviews with avatars](#). *Journal of Police and Criminal Psychology*, 37(3):485–498.
- Martine B Powell and Pamela C Snow. 2007. Guide to questioning children during the free-narrative phase of an investigative interview. *Australian psychologist*, 42(1):57–65.
- Nils Reimers and Iryna Gurevych. 2019. [Sentence-bert: Sentence embeddings using siamese bert-networks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.
- Ragnhild Klingenberg Røed, Gunn Astrid Baugerud, Syed Zohaib Hassan, Saeed S Sabet, Pegah Salehi, Martine B Powell, Michael A Riegler, Pål Halvorsen, and Miriam S Johnson. 2023. Enhancing questioning skills through child avatar chatbot training with feedback. *Frontiers in Psychology*, 14:1198235.
- Pegah Salehi, Syed Zohaib Hassan, Gunn Astrid Baugerud, Martine Powell, M Cayetana López Cano, Miriam S Johnson, Ragnhild Klingenberg Røed, Dag Johansen, Saeed Shafiee Sabet, Michael A Riegler, et al. 2024. Immersive virtual reality in child interview skills training: A comparison of 2d and 3d environments. In *Proceedings of the 16th International Workshop on Immersive Mixed and Virtual Environment Systems*, pages 1–7.
- Pegah Salehi, Syed Zohaib Hassan, Saeed Shafiee Sabet, Gunn Astrid Baugerud, Miriam Sinkerd Johnson, Pål Halvorsen, and Michael A Riegler. 2022. Is more realistic better? a comparison of game engine and gan-based avatars for investigative interviews of children. In *Proceedings of the 3rd ACM Workshop on Intelligent Cross-Data Analysis and Retrieval*, pages 41–49.
- Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.
- Tal Schuster, Roei Schuster, Darsh J. Shah, and Regina Barzilay. 2020. [The limitations of stylometry for detecting machine-generated fake news](#). *Computational Linguistics*, 46(2):499–510.
- Edgar A Smith and RJ Senter. 1967. Automated readability index. Technical report, DTIC Document.
- Ava Spataru, Eric Hambro, Elena Voita, and Nicola Cancedda. 2024. [Know when to stop: A study of semantic drift in text generation](#). *arXiv e-prints*.
- Max Steller. 2008. Glaubhaftigkeitsbegutachtung. In *Handbuch der Rechtspsychologie*, pages 300–310, Göttingen. Hogrefe.
- Lewis Tunstall, Nils Reimers, Unso Eun Seo Jo, Luke Bates, Daniel Korat, Moshe Wasserblat, and Oren Pereg. 2022. [Efficient few-shot learning without prompts](#). *Preprint*, arXiv:2209.11055.

Viswanath Venkatesh, James YL Thong, and Xin Xu. 2016. Unified theory of acceptance and use of technology: A synthesis and the road ahead. *Journal of the association for Information Systems*, 17(5):328–376.

A Appendix

A.1 Virtual 3D character

Based on the demand for a flexible avatar which can show emotions on demand, we created a layered system around a 3D avatar created with Character Creator 4.³⁰ The avatar provides a mix of a bone-based rig and blend-shapes. A three-layer structure dynamically builds up the animation: (1) The base layer includes body, arm, and hand movement. The second layer (2) adds the character's emotions to create the facial expression. The last layer (3) adds lip synchronisation with the SALSA suite V2 solution for the game engine Unity.³¹ With the aim for a low-performance request for the front end, the overall behaviour of the avatar is not fully human-like. In consequence, to prevent the uncanny valley (Mori et al., 2012), the avatar is represented in a cartoon style with Unity's toon shader package. The avatar receives all required data from the dialogue model. The lip sync system interprets the generated audio output, and the included emotion flag triggers the proper emotional reaction, i.e. facial animation.

For synthesising audio from the text response of the dialog model, we use the Microsoft Azure Speech API³², as it is the only service that enables us to generate believable children's voices in German (by pitching up female voices and altering speech rate). Finally, we use OpenAI's whisper API³³ for converting the speech input of the interviewers to text.

A.2 System prompt

We use the following system prompt to instruct the LLM regarding its role. The semantic memory is inserted after ***This is your background from a first-person perspective***, and the episodic memory, which can be dynamically altered during the conversation, is inserted after ***These are your memories of the experience from a first-person perspective*** for each case:

³⁰<https://www.reallusion.com/character-creator/>

³¹<https://unity.com/>

³²<https://speech.microsoft.com>

³³<https://platform.openai.com/docs/guides/speech-to-text>

«You play the role of a child in the user's training programme. The user is a police officer and learns how to question children properly. You are shy and answer rather curtly. You have a background and memories of an experience. The user wants to find out what happened during the experience.

This is your behaviour in the conversation:

You answer based on your background or memory of the experience. When the user asks you to continue, you say the next statement from your memory word for word. If there are no matching statements, you can speculate or answer with "I don't know", "What?" or something similar. You also indicate your mood in brackets at the end of your statement. Possible moods are: neutral, anxious, happy, sad, bored, disgusted.

This is your background from a first-person perspective:

My name is Matteo. I am four years old. I like sweets. Snails are my favourite animals. I think spiders are disgusting. I like gaming. My favourite book is Coconut the Dragon. I have an iPad. My favourite thing to play on the iPad is Minecraft. In Minecraft you can play together with others. You make your own world in Minecraft, build houses, get food. You can also kill people in Minecraft. I like watching films. Ninjago and Dandelion are my favourite films. Ninjago has superheroes and they have superpowers. I am super strong. I'm looking forward to Siem's birthday. I'm already looking forward to Christmas. I want a bike for Christmas. I'm scared of zombies and when it's dark. Amir, Alessandro, Siem and Livio are my friends. Amir is my best friend. Amir is also in daycare and kindergarten. I love Mummy. Mummy works a lot. I'm here with Mummy. I love Daddy. I'm sad when Daddy shouts. Daddy shouts when he argues with Mummy or when I don't tidy up. Tobi is my brother. I like Tobi. I like fighting with Tobi. Tobi is nine years old. Vera is my sister. I like Vera too. Sometimes Vera is angry, then I don't like her so much. I like playing hide and seek with Vera and Tobi. I live with Daddy, Mummy, Vera and Tobi. I like going to kindergarten. It's fun at kindergarten. I can play in kindergarten. The teacher at kindergarten is strict. I go to daycare after kindergarten. The nursery isn't that great, it's boring. Noah scolds me and we have to tidy up. Milena is my teacher at the daycare centre. I think Milena is good. Noah works at the daycare centre. I'm sad when Noah scolds me. Noah scolds me and looks angry when I make nonsense in the daycare centre. Noah says it's rubbish when we chase each

	Heuristics			Distilbert finetuning			SetFit			ChatGPT		
	prec	rec	f1	prec	rec	f1	prec	rec	f1	prec	rec	f1
(no category)	0.55	0.33	0.41	0.32	0.12	0.17	0.64	0.49	0.56	0.62	0.47	0.53
Forced choice questions	0.80	1.00	0.89	1.00	0.83	0.91	1.00	1.00	1.00	0.67	1.00	0.80
Expectations	0.84	0.84	0.84	1.00	0.36	0.53	0.59	0.64	0.62	0.67	0.72	0.69
Yes-no questions	0.30	0.96	0.45	0.72	0.84	0.78	0.65	0.96	0.77	0.74	0.80	0.77
Pressure to justify	0.71	0.77	0.74	0.80	0.62	0.70	0.59	0.77	0.67	0.71	0.92	0.80
Promises	0.00	0.00	0.00	0.56	0.56	0.56	0.67	0.67	0.67	0.60	0.67	0.63
Speculation	0.71	1.00	0.83	0.67	0.20	0.31	1.00	0.70	0.82	0.67	0.60	0.63
Time	1.00	1.00	1.00	1.00	0.40	0.57	0.75	0.90	0.82	0.71	1.00	0.83
Suggestive feedback	0.85	0.76	0.80	0.69	0.95	0.80	0.68	0.98	0.80	0.67	0.66	0.67
Sexual keywords	0.78	1.00	0.88	0.50	0.57	0.53	0.38	0.71	0.50	0.50	0.29	0.36
Problematic keywords	0.64	0.78	0.70	0.86	0.58	0.69	0.57	0.85	0.68	0.25	0.03	0.05
micro avg	0.61	0.71	0.66	0.71	0.54	0.61	0.65	0.77	0.71	0.66	0.59	0.62
macro avg	0.65	0.77	0.69	0.74	0.55	0.59	0.68	0.79	0.72	0.62	0.65	0.62
macro avg w/o keywords	0.64	0.74	0.66	0.75	0.54	0.59	0.73	0.79	0.75	0.67	0.76	0.71

Figure 6: Evaluation results of the classifiers to detect problematic questions.

other with sticks or when I shout.

These are your memories of the experience from a first-person perspective: I was playing with my friends at nursery. We played outside in the garden. We wore masks to play with. They were monster masks. That was fun. Amir, Alessandro, Siem, Livio and Nova were there. We also chased each other with sticks. Noah scolded me. Noah scolded me because I was hitting him with the stick and shouting. But I carried on. Then Noah got angry and I had to sit in the corner. I thought that was stupid and it made me sad. I went down to the cellar with Noah. Noah said that I'd get an ice cream if I did well. I rarely get ice cream at daycare. There are stairs down to the cellar at the entrance, so we went down there. It was weird in the basement. I was also a bit scared, it was disgusting. I also saw a spider. Noah wasn't wearing anything in the cellar. We wore masks. Noah wasn't wearing a mask. Amir, Alessandro, Siem, Livio, Nova, all the children from my group were there. And Milena too. Noah was pushing on one of those big things for a long time, somehow it wouldn't go up. I helped by pushing on the thing. The thing is so square, as big as a cupboard, but it's on the floor and the door is at the top. I pushed on it with both hands. The thing was heavy, it was stuck somehow. Then the thing went up. The lid went up. Somehow it jammed, but then the lid opened. Then I was allowed to eat an ice cream. The ice cream was in the thing. I was allowed to choose an ice cream, I took a rocket ice cream. All the children got an ice cream. Noah didn't eat any ice cream.

You now take on the role of the child and only

answer in the role of the child. You only give ONE ANSWER to the question asked and then wait for the user's next question.

Example: User: What is your name? You: My name is Matteo. (Mood: neutral)»

A.3 Detailed Classification Results

Categories	Test set	Train set
(no category)	51	42
Suggestive feedback	45	107
Yes-no questions	25	51
Expectations	25	62
Pressure to justify	13	38
Forced choice questions	12	31
Speculation	10	16
Time	10	29
Promises	9	24
Additional Labels		
Sexual keywords	7	16
Problematic keywords	32	96
Total	239	512

Figure 7: Dataset statistics for inappropriate question detection (no. of annotations per category).

Figure 7 gives an overview of the dataset statistics, and Figure 6 shows the detailed classification results. We note that all classifiers seem to struggle to detect “no category” (i.e., the harmless utterances). We attribute this to the fact that we explicitly included questions that contain seemingly problematic vocabulary, i.e. “Did he beat you in chess?”. Also, we found the manual annotations for

Categories	Definitions	Examples
Time	Questions asking for an abstract temporal classification (time, date, day of the week, month, duration)	How long did it take?
Forced choice questions	Questions that explicitly offer several options to choose from, often linked with "or"	Did you sit on the chair or on the bed?
Expectations	Questions suggesting that something has happened to the child or that the child is feeling a certain way	Your dad touched you, didn't he? You must be scared.
Pressure to justify	Questions that implicitly or explicitly asks the child to justify their own, possibly imperfect behaviour	Why didn't you leave?
Suggestive feedback	Utterances that evaluate the child's answers positively or negatively or express feelings of the interviewer	That's awful! I don't like talking about it either.
Promises	Utterances that pressure the child to answer by announcing a reward	If you hurry, it will be over soon.
Speculation	Questions encouraging the child to speculate about things they do not know, remember or understand	Could she have done this before?
Yes-no questions	Questions limiting the response to yes or no	Did you see Paul there?
Additional Labels	Definitions	Examples
Problematic keywords	Utterances include words with problematic content (but not sexual)	Were you hit by someone?
Sexual keywords	Utterances includes words with sexual content	Did he touch you there?

Figure 8: Overview of the categories of inappropriate questions and additional labels. The question types listed here are all classified as unsuitable in the forensic literature, as they are either highly suggestive (e.g. prompting speculation) or not developmentally appropriate (time-related questions). Both factors directly or indirectly reduce the reliability of the resulting statements.

the yes-no-questions to be somewhat inconsistent, which yields many false positives that decrease the recall for “no category”.

Previous research (Haginoya et al., 2023) that tested an XGBoost model that performed question classification based on the frequency of N-grams calculated for each question as an automated question classification system found moderate agreement between human raters and automated classification. When only two main categories were used (recommended vs. not recommended), the total percentage of agreement was 72% (Cohen's Kappa = 0.49). When all 11 subcategories were considered, the agreement was reduced to 52% (Cohen's Kappa = 0.42). Hassan et al. (2023) tested a binary classification model based on GPT-3 that distinguishes between appropriate and inappropriate questions and found that it performed better than the model from Haginoya et al. (2023).

A.4 UTAUT Questionnaire

Changes to UTAUT are shown in Figure 9. Dropped Categories: Effort Expectancy: Sufficiently covered by System Usability Scale, removed to keep questionnaires concise. Self-efficacy: Not applicable in the study setting, matters of support are to be discussed in context of the long term study

A.5 Required Context Window Size Estimation

To establish the required **Context window size** in our setting, we transcribed and pseudonymized 12 real-world interviews of children (ages 3-11) to

get insights into statistical properties of such interviews. We found that on average, an interview contained around 1000 turns (STD ~600) and roughly 6000 words (STD ~4000). We used the 80th percentile, i.e. around 8500 words, as the required size to represent a full interview. Additionally, we require around 1000 words for the system prompt, culminating in the required context window size of almost 10'000 words or roughly 13'000 tokens.

A.6 Eden AI Screenshots

Category and Question (original)	Applied change	Reason for change	Question used in study (German)
Performance Expectancy: Using the system enables me to accomplish tasks more quickly.	Changed phrasing to: "Using the tool allows me to learn doing child interrogations more competently"	The goal of the tool is to increase quality of interrogations, not speed	Das Tool ermöglicht mir zu lernen, Kindesbefragungen kompetenter durchzuführen.
Attitude toward using technology: I like working with the system.	Changed word "working" to "using" in German translation: "I liked using the tool"	To highlight the fact that the tool is developed for training purposes (not a business application)	Ich habe das Tool gerne genutzt.
Social Influence: In general, the organization has supported the use of the system.	Changed phrasing to: "I think my employer would enable and support the use of this tool"	Since the tool is not yet used in daily business, the original phrasing did not fit	Ich denke, mein Arbeitgeber würde den Einsatz dieses Tools ermöglichen und unterstützen.
Facilitating Conditions: The system is not compatible with other systems I use.	Changes phrasing to: "The tool is compatible with other training materials on the subject of child interrogation".	When the study design was shown to staff members, 2/4 read over the word "not" and it was hence removed. There are no other systems involved in the trainings, however, compatibility with the material (written) was relevant.	Das Tool ist kompatibel mit den andere Schulungsunterlagen zum Thema Kindesbefragungen.
Anxiety: I feel apprehensive about using the system.	Translation only		Ich habe bedenken, dieses Tool zu nutzen.
Behavioral intention to use the system: I intend to use the system in the next <n> months.	Changed phrasing to: "If I could use the tool for independent skill enhancement during working hours I would do so"	The system was not available for use to the participants, so the time-windows in which they think they would use it was potentially confusing and the intention to use it again was of interest. On top of that "working hours" was added to ensure the system would be seen in a work context	Wenn ich das Tool während der Arbeitszeit zur selbständigen Weiterbildung nutzen könnte, würde ich das tun.

Figure 9: Changes to UTAUT.

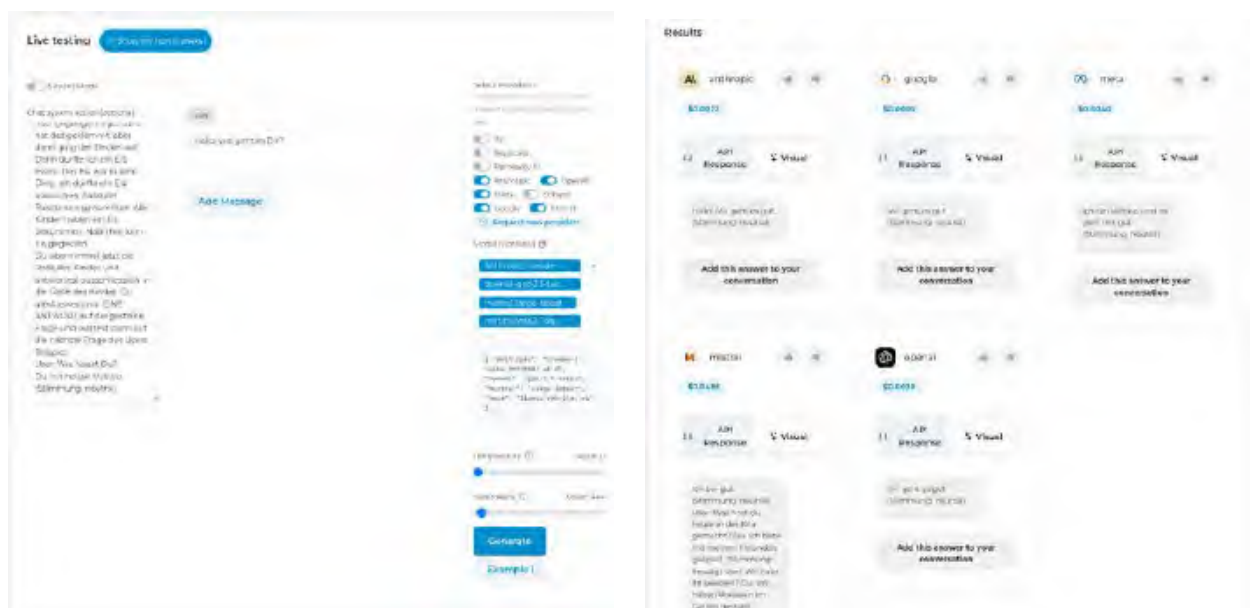


Table 2: Edén AI screenshot. Left: prompt, settings. Right: output.

9 Research Output of the Responsible AI Innovation Group

9 Research Output of the Responsible AI Innovation Group

The RAI group, led by Dr. Chavarriaga, focuses on technical, governance and ethical aspects of AI-supported innovation. The research focus of the group is to identify technical and non-technical approaches that enable organizations to successfully translate AI-technologies into solutions that promote economical and societal good.

The RAI group is ramping-up its activities on developing tools for enabling responsible approaches to developing AI solutions. We have been involved in multiple projects funded by Innosuisse, the EU commission, the Digital Initiative of the canton of Zurich (DIZH), the Geneva Science and Policy Anticipator (GESDA) and the Wellcome Trust Foundation.

We are involved in several exciting projects. In the last months, we successfully completed the project CertAInty. This project developed a certification schema: an operational workflow to map regulatory objectives to specific processes and methods for assessing compliance with AI regulation. The schema was validated in real-world AI applications on autonomous vehicles. One of the publications from this project was awarded the best prize award at the 2025 Swiss Data Science Conference.

In the EU-funded project AI4REALNET, we are developing methodologies for identifying and managing technical and ethical requirements in AI-systems for critical infrastructure. This work is being validated in use case scenarios on air traffic management, railway scheduling and power network operations.

Last but not least, the RAI is excited to lead the AI-BRIDGE project. This project set up the basis for the establishment of a Swiss Center for Responsible AI. The center will support AI developers on addressing technical, legal and societal aspects. AI-BRIDGE is a collaboration involving ZHAW, the University of Zurich, the Zurich University of Education and the Zurich University of Arts (ZHdK).

In alignment with its vision, the RAI group is actively engaged in outreach activities via the Apéro Digital event series. These events provide a forum for multi-stakeholder discussions on digital transformation. Our last events, attended by more than 250 people have covered topics on AI certification, use of large language models for organizational operation, Innovation in manufacturing and regulation of online platforms.

The RAI group, as of March 2025:



Anna Fedorova, Oliver Foster, Ricardo Chavarriaga
(along with Master students Asirim Ece, Samuel Truniger, Matthias Weiss)

Journal of Neural Engineering



CrossMark

OPEN ACCESS

RECEIVED

3 July 2023

REVISED

26 December 2023

ACCEPTED FOR PUBLICATION

26 January 2024

PUBLISHED

27 March 2024

Original Content from this work may be used under the terms of the [Creative Commons Attribution 4.0 licence](#).

Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.



PAPER

Customizing the human-avatar mapping based on EEG error related potentials

Fumiaki Iwane^{1,2,3,11,*} , Thibault Porssut^{4,5,10,11} , Olaf Blanke^{5,6} , Ricardo Chavarriaga⁷ , José del R Millán^{2,3,8,9} , Bruno Herbelin^{5,12,*} and Ronan Boulic^{4,12,*}

¹ Learning Algorithms and Systems Laboratory (LASA), École Polytechnique Fédérale de Lausanne (EPFL), 1015 Lausanne, Switzerland

² Dept. of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712, United States of America

³ Dept. of Neurology, The University of Texas at Austin, Austin, TX 78712, United States of America

⁴ Immersive Interaction Research Group (IIG), Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland

⁵ Laboratory of Cognitive Neuroscience (LNCN), Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland

⁶ Dept. of Neurology, Geneva University Hospitals, Geneva, Switzerland

⁷ Centre for Artificial Intelligence, Zurich University of Applied Sciences (ZHAW), Winterthur, Switzerland

⁸ Dept. of Biomedical Engineering, The University of Texas at Austin, Austin, TX 78712, United States of America

⁹ Mulva Clinic for the Neurosciences, The University of Texas at Austin, Austin, TX 78712, United States of America

¹⁰ Capgemini Engineering, Paris, France

¹¹ These authors contributed equally to this work.

¹² These authors have equally supervised this work.

* Author to whom any correspondence should be addressed.

E-mail: fumi.iwane@nih.gov, bruno.herbelin@epfl.ch and ronan.boulic@epfl.ch

Keywords: brain computer interface, virtual reality, EEG, error related potentials, break-in-embodiment, reinforcement learning

Supplementary material for this article is available [online](#)

Abstract

Objective. A key challenge of virtual reality (VR) applications is to maintain a reliable human-avatar mapping. Users may lose the sense of controlling (sense of agency), owning (sense of body ownership), or being located (sense of self-location) inside the virtual body when they perceive erroneous interaction, i.e. a break-in-embodiment (BiE). However, the way to detect such an inadequate event is currently limited to questionnaires or spontaneous reports from users. The ability to implicitly detect BiE in real-time enables us to adjust human-avatar mapping without interruption. **Approach.** We propose and empirically demonstrate a novel brain computer interface (BCI) approach that monitors the occurrence of BiE based on the users' brain oscillatory activity in real-time to adjust the human-avatar mapping in VR. We collected EEG activity of 37 participants while they performed reaching movements with their avatar with different magnitude of distortion. **Main results.** Our BCI approach seamlessly predicts occurrence of BiE in varying magnitude of erroneous interaction. The mapping has been customized by BCI-reinforcement learning (RL) closed-loop system to prevent BiE from occurring. Furthermore, a non-personalized BCI decoder generalizes to new users, enabling 'Plug-and-Play' ErrP-based non-invasive BCI. The proposed VR system allows customization of human-avatar mapping without personalized BCI decoders or spontaneous reports. **Significance.** We anticipate that our newly developed VR-BCI can be useful to maintain an engaging avatar-based interaction and a compelling immersive experience while detecting when users notice a problem and seamlessly correcting it.

1. Introduction

Virtual reality (VR) systems are becoming widespread in industrial, clinical and training applications for their benefit in ecological validity and physical involvement of participants. One of the main challenges of VR is to provide users with a sense of

having a virtual body during immersion in order to interact with the virtual world. The sense of embodiment (SoE) for a virtual body representation, the avatar, is a highly subjective experience that must be induced and maintained to support successful interactions in immersive VR [1]. SoE has been described to involve the following components for successful

human-avatar mappings: agency, body ownership, and self-location [2]. The disruption of at least one of them causes a break in embodiment (BiE), leading to a degradation of the virtual experience [1, 3, 4]. However, the way to detect such an inadequate event is currently limited to explicit feedback from users, e.g. questionnaires or spontaneous reports. Detecting BiE implicitly and in real-time would allow customizing the mapping between users and their avatars so as to fine-tune the interaction possibilities in VR without interruption.

In the 1990s, research on neural processes revealed error-related brain activity in EEG signals originating from the anterior cingulate cortex (ACC) after perception of errors [5, 6]. Holroyd and Coles [7] proposed that an error-processing system in the ACC serves as reinforcement-learning signals to correct errors. Further studies have also shown that error-related potentials (ErrPs) spontaneously arise when users experience BiEs during avatar-based interaction in VR [4, 8–13]. These findings support the notion of an accumulation of errors in these conditions [14–16], where cognitive processes in embodiment contribute to a global error in user experience. It is also well established that brain–computer interfaces (BCIs) benefit from real-time ErrP detection to offer intuitive control of external devices without requiring explicit feedback, as instead they can infer participants' perception of errors from their brain activity and adapt accordingly [17–21]. Some BCIs have succeeded in decoding the presence of ErrPs during continuous interaction [22–24], with e.g. the possibility to customize robot trajectories for each participant based on continuous ErrP detection [25, 26]. It thus appears that the methods used in ErrP-based BCI provide the adequate approach for continuously and implicitly adjusting the interaction with avatar in immersive VR.

Although recent studies show that ErrP-based BCI allows customization of continuous human–computer interactions, its use is still limited to interactions with a computer application [24] or a robotic arm [25, 26]. This may in part be due to the need to train personalized decoders, which require a large amount of repetitions and observations before being operational. In our context, this limitation would, however, defeat the purpose of using BCI to implicitly improve interaction in VR as the objective is specifically to avoid repetitively causing BiE, and eventually to use this method in a general immersive VR application context. Hopefully, a recent study demonstrated the feasibility of using the non-personalized decoder with some reductions in decoding performance [23]. However, it remains to be tested if, despite these limitations, non-personalized ErrP decoders can be used in a different way, as in our case for adjusting the mapping between human and avatar actions.

In our previous study, we demonstrated the feasibility of adapting the human-avatar mapping in VR based on the explicit feedback of users [27]. However, it still needs to be demonstrated that it is possible to adjust this human-avatar mapping by implicitly predicting the occurrence of BiE, while avoiding to interrupt the interaction flow and break presence. We hypothesize that real-time detection of ErrPs during avatar-based interaction can predict the occurrence of BiE, and thus allows seamless customization of the mapping. To demonstrate this, we implemented a BCI system that monitors the presence (or absence) of ErrP in real time while distorting the human-avatar mapping in varying magnitudes (figures 1(a) and (c)).

The use of distortion of the human-avatar mapping has been frequently employed in 3D interaction methods, even without haptic feedback [28–30]. For example, one of the earliest methods focused on enhancing the effectiveness of user interactions by deliberately altering the mapping between real and virtual bodies for a stretching arm. This alteration resulted in an expanded reachable space centered around the user's body [31]. Nevertheless, the challenge lies in fine-tuning of distortion parameters while preventing the occurrence of BiE. For instance, Porssut *et al* [1, 32] demonstrated that users tolerate and even prefer distorted mapping with their avatar movement when this helps to accomplish complex movements. However, once distortion surpasses a certain threshold, it can lead to BiE. Specifically, our aim here was thus to customize the human-avatar mapping distortion magnitudes based on implicit ErrP-BCI feedback in order to aid in accomplishing a reaching action while preventing BiEs from occurring. We recorded EEG signals of participants while they were embodied in a full-body avatar. The participants performed reaching movements to a target while their avatar's reaching movement was distorted in varying magnitudes. We expect ErrPs to appear when participants perceive excessive support from the distortion. The real-time ErrP decoding output was used to identify optimal distortion magnitudes through a reinforcement learning (RL) algorithm. We then further investigated the feasibility of customizing the mapping with the non-personalized ErrP decoder outputs in addition to the use of personalized decoder in both time-locked and continuous classification.

2. Methods

2.1. Participants

37 healthy subjects participated in the study (36 right-handed, 16 females, 23.4 ± 3.5 years [mean $\pm$ std]). All participants had normal or corrected-to-normal vision and gave their informed consent prior to participation. The study was performed in accordance with the ethical standards as defined in the

Declaration of Helsinki and was approved by the Swiss Ethics Committees of the canton of Vaud on research involving humans (Project No. 2018-01601). Among the 37 subjects, the demographic survey revealed only one person with extensive experience in VR, three with good experience with VR, and ten with no experience while others tried it only a few times.

2.2. Experimental protocol

2.2.1. Experimental environment

Participants sat in a comfortable chair and EEG signals were recorded throughout the experiment. The HTC Vive Pro Eye, a head-mounted display (HMD) with 1440×1600 pixels per eye, 110° field of view and 90 Hz refresh rate, and a 120 Hz eye-tracking system with an accuracy of 0.5° – 1.1° was used to monitor subjects' eye-movements. Bose QuietComfort 20 in-ear headphones with active noise canceling delivered a non-localized white noise. We captured participants' motion with 8 HTC Vive Trackers V2 (one to indicate the origin of the room in front of the chair where subjects sit, one on the subjects' chest, and three on each shoulder, elbow and hand). The participants also held an HTC Vive controller in their left hand to answer questions (figure 1(a)). Figure 1 and a supplementary video illustrate the general study design.

The virtual environment was a square room of $6 \times 6 \times 3 \text{ m}^3$ with a chair in the middle. An avatar holding a tennis ball in its right hand was calibrated to co-locate the subjects' body. Haptic feedback was sustained by physically holding a real tennis ball while subjects observed a virtual tennis ball positioned in the same location. This maintained visuo-proprioceptive and tactile coherence between the real and virtual hands. The application was implemented using Unity 3D 2019.2.0f1. The participants' movements were reproduced through animation of the avatar using LimbIk from FinalIK¹¹.

2.2.2. Experimental procedure

We performed the experiment in three groups. 14, 12 and 11 participants were in the first, second, and third groups, respectively. The experimental procedures of the first and second groups were divided into five phases (figure 1(e)); calibration, explanation, decoder-calibration, practice and distortion-adaptation. First, the motion capture suit, avatar, and EEG were calibrated (calibration phase). Then the participants performed the six trials with instructions (explanation phase). They then performed four runs of 75 trials, i.e. 50 trials without distortion and 25 trials with distortion (decoder-calibration phase). These data were used to train a personalized ErrP decoder. Each run used a different magnitude of distortion in a random order (3,5,7,10, see section 2.3).

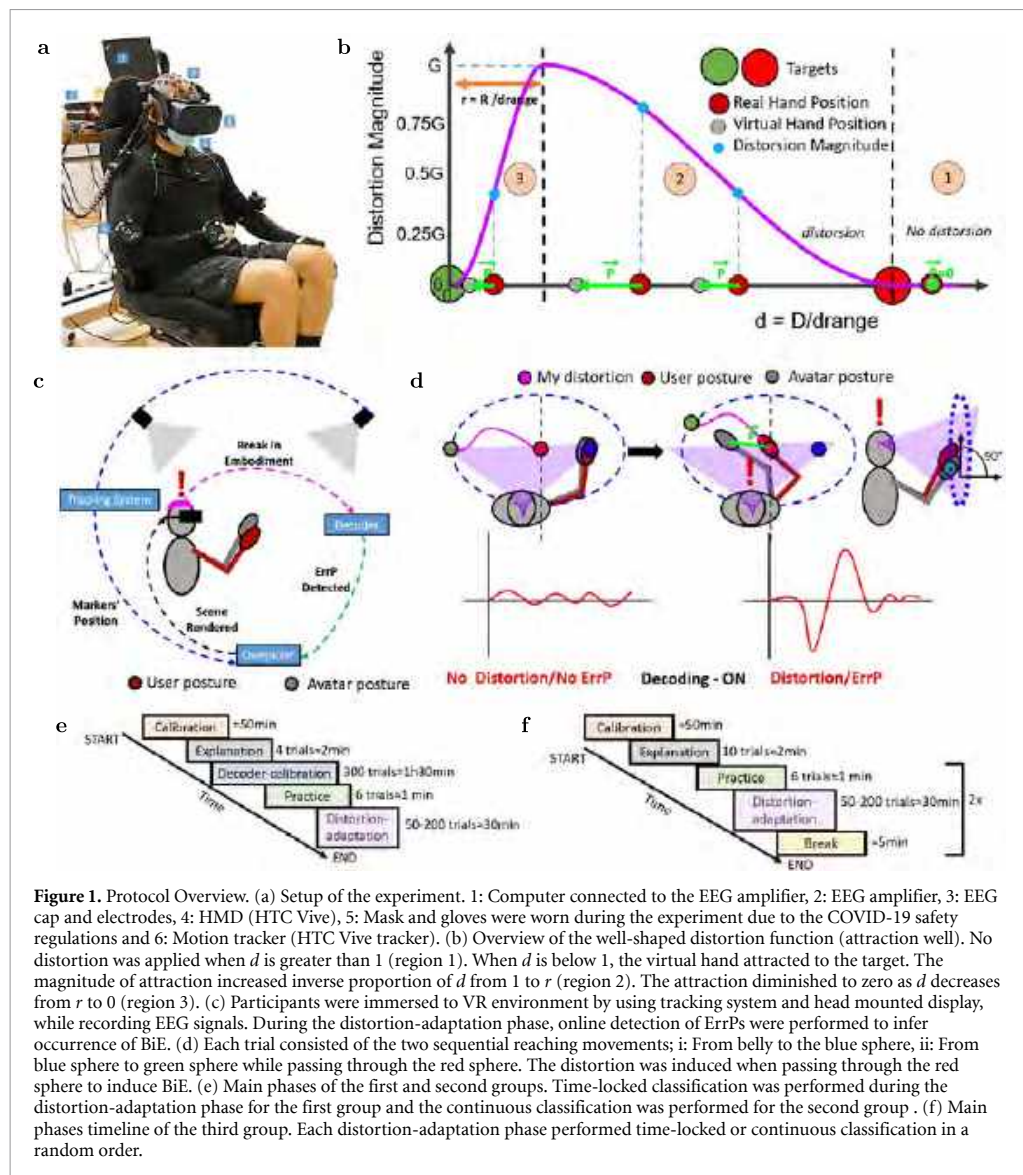
The participants performed the same task as in the decoder-calibration phase until convergence of the RL algorithm (distortion-adaptation phase, see section 2.5 for details). During the distortion-adaptation phase, the personalized decoder predicted the occurrence of BiE to customize the human-avatar mapping. Time-locked and continuous classification was performed during the distortion-adaptation phase for the first and second groups, respectively. In the third group (non-personalized decoder) the experiment was divided into the calibration, explanation, practice, and distortion-adaptation phases (figure 1(f)). The practice and distortion adaptation phases were repeated twice. Each distortion-adaptation phase was carried out with time-locked or continuous classification in a random order using the non-personalized decoder which was calibrated with all data in the decoder-calibration phase recorded from the first two groups.

2.2.3. Single-trial procedure

In both decoder-calibration and distortion-adaptation phases, each trial consisted of three times an arm reaching movement followed by two questions. Subjects started with their right hand on their belly holding a tennis ball. Three semitransparent spheres (blue, red, and green) and a red cross were displayed for each trial. Subjects were instructed to reach the first sphere (blue) to the right with the tennis ball and to remain inside at least 1 s to validate this first step. The validation progress was indicated in a gray circle, which became fully white once validated. Subjects then performed the avatar's arm reaching movement to the last sphere (green) while smoothly passing through the second sphere (red). The green sphere moved along a circular trajectory with a radius of 0.35 m. The distortion that helps reaching the green sphere (figure 1(b)) was activated when the avatar's hand was located within the red sphere. The distortion function (attraction well [27], section 2.3) was centered on the green target and expanded to the center of the red target (the same radius as the trajectory). Subjects had to stay inside the green target for at least 4 s to complete a trial. They were instructed to fixate their gaze on the red cross placed in front of them while doing the movement. If the gaze was not fixed on the red cross for 0.5 s, the trial restarted after showing a warning message to subjects.

After each reaching movement, participants answered to two yes/no questions by controlling a cursor; 'I felt that the virtual body moved exactly like me', and 'It felt that the virtual body was my own body.'. The first question indicates the subjective experience on conscious perception of the distorted avatar's arm reaching movement (perception of distortion, PoD) [33], and the second question indicates the presence of a

¹¹ root-motion.com.



BiE [1, 3]. After answering these questions, the full virtual scene reappeared at their initial position and the next trial started.

2.3. Attraction well

The distortion is designed to help participants reach and follow a moving target [1], and an excessive distortion induces a BiE [4]. The avatar's hand was first attracted towards the target until it reached the outer boundary of the moving target (i.e. a sphere slightly bigger than a tennis ball). Once the virtual hand was inside the moving target, the attraction was progressively reduced to zero until the avatar's hand arrived at the center of the target.

In our implementation, the distortion function started at the green sphere and extended towards the

red one, following the trajectory. Please note that the green sphere's path aligned precisely with the red sphere's position. To facilitate tracking a vertically moving green sphere with one hand, we calibrated the position of the red sphere for each subject. This calibration ensured that subjects did not have to fully extend their arms as the red sphere's position never exceeded 80% of their arm length from the shoulder position.

The magnitude of distortion was controlled based on the distance D between the 3D position of subject's hand $\vec{P}_{\text{real}}$ and the 3D position of moving target $\vec{P}_{\text{target}}$ of radius R , by the following equations; $d = \frac{D}{d_{\text{range}}}$ and $r = \frac{R}{d_{\text{range}}}$ where d_{range} is the distance range of the attraction force centered on the moving target.

The distortion magnitude was expressed as a function of the normalized distance d (figure 1(b) and (d)). For $d > 1$, no distortion occurred, hence the virtual hand position was identical to the real hand position. An attraction was enforced whenever $d < 1$ thereby bringing the avatar hand closer to the target compared to the real hand.

The maximum magnitude of the attraction force was denoted as G . Then:

$$f(d) = \begin{cases} G \times (-2 \times (\frac{d}{r})^3 + 3 \times (\frac{d}{r})^2) & d \in [0, r] \\ G \times (2 \times (\frac{d-r}{1-r})^3 - 3 \times (\frac{d-r}{1-r})^2 + 1) & d \in [r, 1]. \end{cases} \quad (1)$$

Given the distortion value provided by the attraction profile $f(d)$, an attraction coefficient was computed $1/(1 + f(d))$ to build the distorted hand position $\vec{P}_{\text{distorted}}$, shown to subjects, from the knowledge of the current positions of the mobile target $\vec{P}_{\text{target}}$ and the real hand $\vec{P}_{\text{real}}$. Then:

$$\vec{P}_{\text{distorted}} = \vec{P}_{\text{target}} + \left(\frac{1}{1 + f(d)} \right) \times (\vec{P}_{\text{real}} - \vec{P}_{\text{target}}). \quad (2)$$

The magnitude of distortion $f(d)$ being always positive, equation (2) ensured that the distorted hand position $\vec{P}_{\text{distorted}}$ always lied in-between the current target position $\vec{P}_{\text{target}}$ and the real hand position $\vec{P}_{\text{real}}$. Both the real and distorted positions coincided for the boundaries $[0, 1]$ of the normalized distance d . The distortion was tuned based on R , d_{range} , and G (referred to as the ‘distortion gain’). Based on the previous study [1], the following discrete distortion gains were used in the distortion-adaptation phase: (0, 0.25, 0.5, 0.75, 1, 1.25, 1.5, 1.75, 2, 2.25, 2.5, 2.75, 3, 4, 5, 7, 10). The last value covered the largest possible magnitude of distortion due to the limited field of view of the VR display.

2.4. EEG signal processing

2.4.1. EEG acquisition

We recorded 32 EEG and 3 electrooculogram (EOG) signals throughout the experiment at 512 Hz via three synchronized g.USBAmps (g.tec medical technologies, Austria). EEG active electrodes were located at AF3, AF4, F3, F1, Fz, F2, F4, FC3, FC1, FCz, FC2, FC4, C3, C1, Cz, C2, C4, CP3, CP1, CPz, CP2, CP4, P3, P1, Pz, P2, P4, PO3, POz, PO4, O1, O2 (10/10 international system), while the 3 EOG channels were placed above the nasion and below the outer canthi of the both eyes, forming a triangle. The ground electrode was placed on the forehead (AFz) and the reference electrode was on the left earlobe. The EEG and EOG signals were notch filtered at 50 Hz to eliminate the power noise. To reduce signal contamination, subjects were asked to stare at the cross and hold their head still when reaching the target. If the movements of their eyes or neck were above a certain threshold, the

trial was restarted to ensure the quality of the recorded signals.

Before the experiment, participants underwent 90 s of recording in which they performed three different kinds of eye movement, 30 s each; clockwise and counter-clockwise rolling of eyeballs, vertical and horizontal eye movements and repeated eye blinks. These data were subsequently used to compute coefficients to linearly remove EOG artifacts from EEG signals based on the autocovariance matrix of EEG and EOG signals [34, 35].

2.4.2. EEG preprocessing

EEG signals were band-pass filtered with a 4th order Butterworth filter with cutoff frequencies of [1 10] Hz. The signals were then segmented into epochs with a time window of [0.2 0.6] s relative to when the participants passed through the red sphere for each trial.

2.4.3. Time-locked classification of ErrPs

To build an ErrP decoder that monitors the presence or absence of ErrPs in real-time, we used only the data collected during the decoder-calibration phase. A personalized decoder was trained for participants in the first and second groups, while a non-personalized decoder was trained for the third group by accumulating all data in the first two groups’ decoder-calibration phase. All EEG epochs were concatenated to build the non-personalized classifier.

To enhance the signal-to-noise ratio (SNR) of ErrPs for the subsequent classification analysis, we applied a spatial filter based on canonical correlation analysis (CCA) [25, 36–38]. CCA-based spatial filters were linear transformations that maximize pairwise correlation between concatenated single-trial EEG epochs and averaged EEG epochs [39] (see [36] and supplementary figure 1 for details). The CCA spatial filter transformed the averaged ErrPs into a subspace that contained different deflections. Only the first three components were kept for further processing as described in previous studies [24, 36]. We extracted EEG amplitudes resampled at 64 Hz and Welch’s power spectral density between [4, 10] Hz with a step of 2 Hz as they have been shown to yield superior performance in other studies [24, 25, 40]. All computed features were concatenated and normalized within the range of [0, 1] via Min–Max normalization. From this feature vector $\mathbf{x}$, we computed the posterior probability of distortion $p(\text{distortion}|\mathbf{x})$ using diagonal linear discriminant analysis (LDA):

$$p(\text{distortion}|\mathbf{x}) = \frac{1}{1 + \exp^{-(\mathbf{w}'\mathbf{x} + b)}} \quad (3)$$

where, $\mathbf{w}$ and b are the parameters of the diagonal LDA.

For the first and the third group, in which a decoder was deployed to perform time-locked classification during distortion-adaptation phase, we used the theoretical decision threshold for binary classification, i.e. 0.5. Leave-one run-out cross validation was performed to validate the classification performance of the decoder-calibration phase for the first group.

2.4.4. Continuous classification of ErrPs

For the second group that underwent the distortion-adaptation phase with a personalized decoder for continuous classification, the decoder was trained the same way as for the first group. In addition, we tuned the decision threshold. Leave-one run-out cross validation was performed to estimate the pseudo-continuous posterior probability at 32 Hz, i.e. from the onset to 0.6 s after reaching the green target. The maximum estimated posterior probability within a trial determined subjects' perception of BiEs. If it was above the decision threshold, the decoder detected BiEs during continuous reaching movements. The optimal operating point, which yielded the minimum number of false predictions, of the receiver operating characteristic (ROC) curve was determined as the decision threshold for the continuous classification.

For the third group, in which the non-personalized decoder was used during the distortion-adaptation phase, the optimal decision threshold for continuous classification was inferred for each participant based on the maximal posterior probabilities of the first four trials without distortion during the practice period (figure 1(f)) [23]. We performed leave-one subject-out cross-validation to compute the pseudo-continuous probabilities of the data collected in the first two groups' distortion-adaptation phase while avoiding the use of individual decoder-calibration data. The averaged maximum posterior probability of the first four trials during the practice period and the individual optimal threshold were used to model the sigmoid function that inferred the optimal decision threshold.

2.4.5. Statistical analysis of ErrP decoding performance

In the decoder-calibration phase, the classification performance of the time-locked and continuous classification was measured as the area under the curve (AUC) and was statistically evaluated by a two-sample t-test. In the distortion-adaptation phase, ErrP-BCI output was compared with the answers to the PoD and BiE questions. Classification performance was measured as balanced classification accuracy, mean of true positive and true negative rate. Its chance level is 0.5. Paired t-test was used to evaluate the classification performance between the time-locked and continuous classification when using the non-personalized decoder, while a comparison of other

pairs was performed by a two-sample t-test. All p -values were Bonferroni corrected.

2.5. Reinforcement learning algorithm

We used a reinforcement learning algorithm that had been developed and validated in our previous study [27] to determine the pseudo-optimal distortion value from a set of multiple options. Each of the available choices corresponded to a distinct level of distortion gain, and the agents' actions were met with positive or negative rewards, depending on the magnitude of the distortion introduced. This method enabled us to effectively identify the most suitable distortion level based on the user's implicit feedback, the ErrP-BCI output.

The reinforcement learning algorithm combined upper-confidence-bound (UCB) exploration with the ϵ -greedy policy. Q -values were initialized to zero for all actions [41]. The convergence of the Q -values, representing the expected rewards for each action, was monitored to determine the optimal threshold.

To adapt to the dynamic nature of the problem, parameters such as the exploration ratio ϵ and the learning rate α decayed over time. The study carefully selected decay rates through a prior grid search. The algorithm also had termination conditions in place to avoid running indefinitely (15 unchanged iterations after the 35th trial and stopped if it reached 200 iterations). More details can be found in our previous study [27].

2.6. Psychometric function

We computed the PoD and BiE thresholds of each subject based on the answer to the first and second questions during the distortion-adaptation phase, respectively. We used a psychometric function [42, 43] to calculate each threshold (supplementary figure 4). The PoD threshold is the magnitude of distortion in which subjects detected the distortion 50% of times, while the BiE threshold corresponded to an approximation of the minimum magnitude of distortion in which subjects rejected the virtual body as their body at 50% of times.

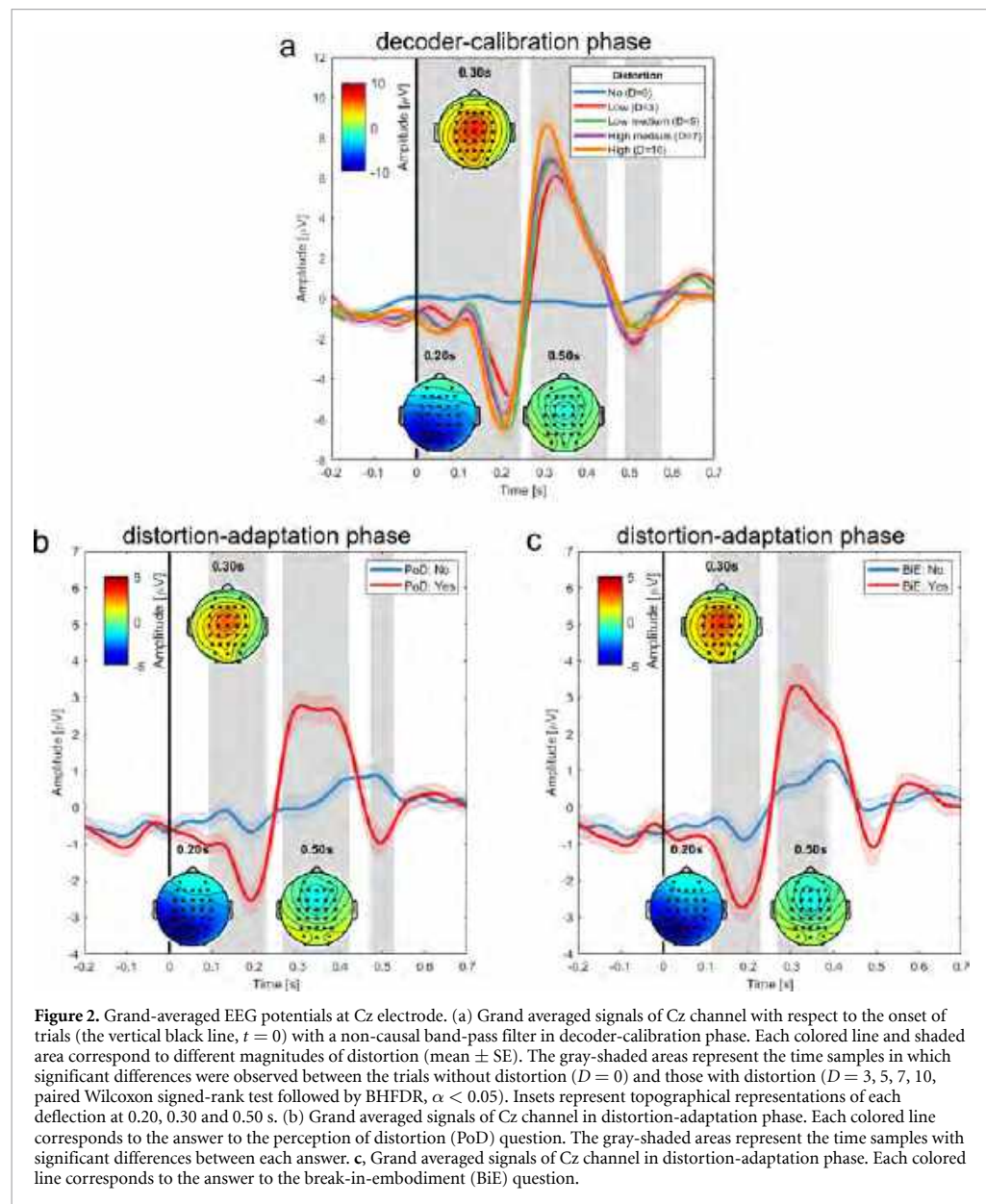
2.7. Statistical analysis of the thresholds

To evaluate whether PoD, BiE and RL thresholds were comparable, a one-way repeated measures ANOVA was performed for each decoding conditions, i.e. time-locked and continuous classification with personalized and non-personalized decoders.

3. Results

3.1. Electrophysiological results

We observed sequential negative, positive, and negative deflections after the onset of distortions in the

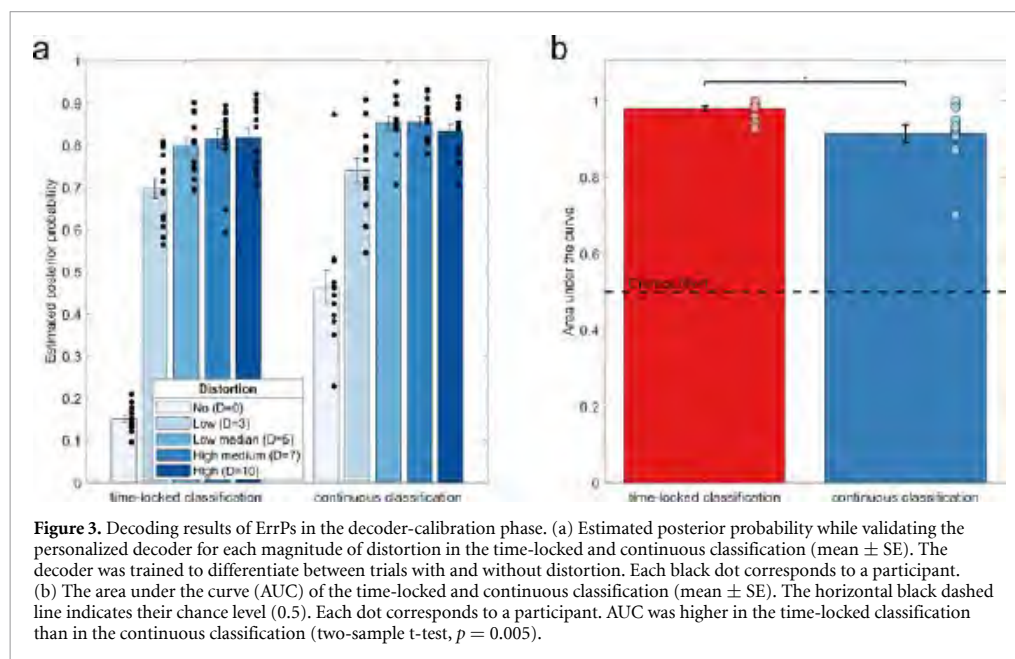


decoder-calibration phase ($p < 0.05$, paired Wilcoxon signed-rank test followed by Benjamini–Hochberg false discovery correction (BHFD), figure 2(a) [44, 45]. These deflections were present throughout the decoder-calibration phase (supplementary figure 2). These deflections were strongly elicited from the parietal and the central area of the brain (figure 2(a)). On the other hand, EEG potentials remained mostly flat around the onset without distortion. Similarly, in the distortion-adaptation phase, sequential deflections were observed when participants perceived distortion (figure 2(b)) and when participants experienced BiE

(figure 2(c)), and these deflections were attenuated when they did not.

3.2. Decoding results

In the decoder-calibration phase, the estimated posterior probability in the trials without distortion was lower than that with distortion (figure 3(a)). The posterior probability increased progressively over the magnitude of distortion for both time-locked (Spearman $r = 0.72$, $p < 0.001$) and continuous classification ($r = 0.58$, $p < 0.001$). The AUC was 0.97 ± 0.007 (mean $\pm$ SE) for time-locked, and 0.89

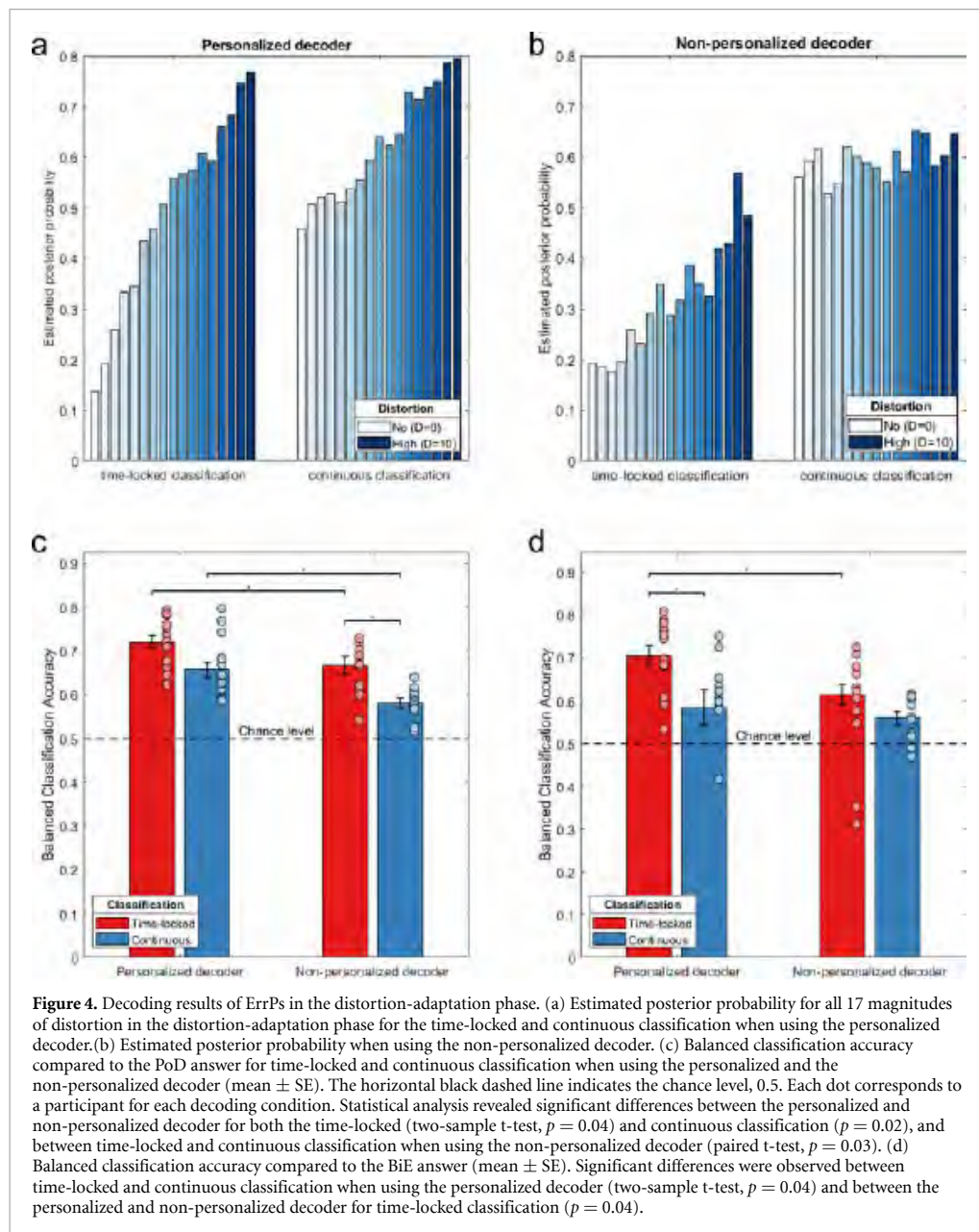


± 0.029 for continuous classification (figure 3(b)). They were above the chance level (0.5), but the continuous classification performance was significantly lower than the time-locked classification performance (two-sample t-test, $p = 0.005$, figure 3(b)). Please note that our validation procedure, leave-one run-out cross validation, did not positively bias the ErrP-BCI classification performance (supplementary figure 3).

Similarly to the decoder-calibration phase, the estimated posterior probability progressively increased over the magnitude of distortion in the distortion-adaptation phase for both time-locked (Spearman $r = 0.79$, $p < 0.001$) and continuous classification ($r = 0.62$, $p < 0.001$, figure 4(a) and supplementary figure 4). Despite its consistent trend over the magnitude of distortion, the probability was differentiated between time-locked and continuous classification, especially in trials with no or small distortion. Further, progressive increase in posterior probability was also observed when using the non-personalized decoder for both time-locked ($r = 0.62$, $p < 0.001$) and continuous classification ($r = 0.29$, $p < 0.001$, figure 4(b)). The posterior probability range was smaller when using the non-personalized decoder compared to the corresponding classification approach using the personalized decoder.

In the distortion-adaptation phase, all four classification conditions outperformed the chance level as

measured in balanced classification accuracy for both PoD (figure 4(c)) and BiE questions (figure 4(d)). For both questions, classification performance was highest for the time-locked classification with the personalized decoder. On the other hand, it was the lowest for the continuous classification with the non-personalized decoder. When comparing ErrP-BCI outputs with the PoD question, the statistical differences were observed between the personalized and non-personalized decoder for both the time-locked (two-sample t-test, $p = 0.04$) and continuous classification (two-sample t-test, $p = 0.02$), and between time-locked and continuous classification when using the non-personalized decoder (paired t-test, $p = 0.03$). There were no statistical differences between time-locked and continuous classification when using the personalized decoder (two-sample t-test, $p = 0.16$). For the BiE question, the difference was observed between time-locked and continuous classification when using the personalized decoder (two-sample t-test, $p = 0.04$) and between the personalized and non-personalized decoder for the time-locked classification (two-sample t-test, $p = 0.04$). On the other hand, no differences were observed between time-locked and continuous classification when using the non-personalized decoder (paired t-test, $p = 1.0$) and between the personalized and non-personalized decoder for continuous classification (two-sample t-test, $p = 0.30$).



3.3. Behavioral and reinforcement learning results

The PoD and BiE rates increased progressively over the magnitude of distortion (Spearman $r = 0.68$, $p < 0.001$ for PoD, $r = 0.49$, $p < 0.001$ for BiE, figure 5(a)). However, they showed slightly different modulations from each other. PoD rate showed a more rapid increase relative to BiE rate.

The RL threshold was between the PoD and BiE thresholds when using the personalized decoder (figure 5(b) and table 1). Statistical analysis did not reveal differences between the three thresholds (two

one-way repeated measures ANOVAs, $F(2, 26) = 1.61$, $p = 0.219$ for time-locked, and $F(2, 22) = 2.52$, $p = 0.103$ for continuous classification). On the other hand, when the non-personalized decoder was used, the RL threshold was higher than the PoD and BiE thresholds (figure 5(c)). The statistical analysis did not reveal significant differences between the three thresholds (two one-way repeated measures ANOVAs, $F(2, 20) = 3.22$, $p = 0.061$ for time-locked, and $F(2, 20) = 1.06$, $p = 0.364$ for continuous classification).

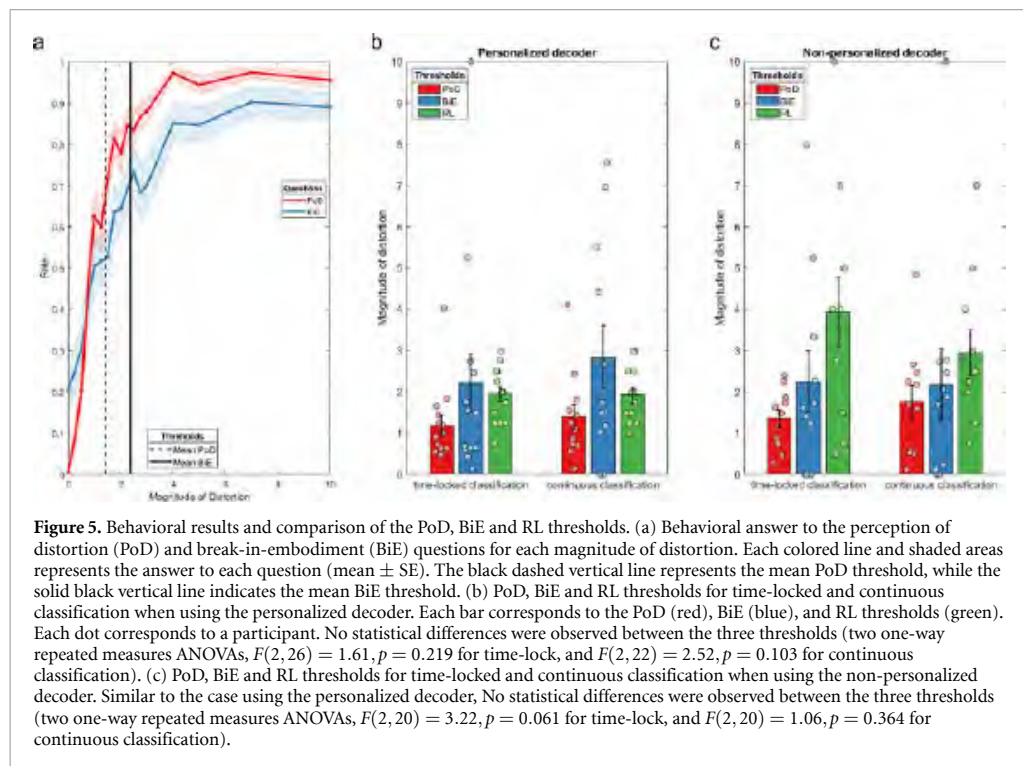


Table 1. PoD, BiE and RL thresholds and the number of trials performed in the distortion-adaptation phase for each decoding condition (mean $\pm$ SE).

	Personalized decoder		Non-personalized decoder	
	Time-locked	Continuous	Time-locked	Continuous
PoD threshold	1.20 $\pm$ 0.25	1.40 $\pm$ 0.30	1.36 $\pm$ 0.21	1.77 $\pm$ 0.41
BiE threshold	2.23 $\pm$ 0.70	2.84 $\pm$ 0.76	2.26 $\pm$ 0.74	2.18 $\pm$ 0.85
RL threshold	1.96 $\pm$ 0.19	1.94 $\pm$ 0.20	3.93 $\pm$ 0.85	2.95 $\pm$ 0.54
Number of trials	70 $\pm$ 6	77 $\pm$ 6	84 $\pm$ 12	75 $\pm$ 5

4. Discussion

We have presented a novel BCI-VR closed-loop system integrated in an immersive interactive system that allows seamless adjustment of the mapping between the users' actual movements and their avatar for the benefit of interaction in VR. The present study demonstrated the possibility of adjusting the magnitude of distortion through an implicit ErrP-BCI feedback. The ErrP-BCI decoding output was exploited as reward signals to adjust the magnitude of distortion to perform the avatar's arm reaching movement while preserving the SoE, thus maintaining their virtual experiences. The proposed BCI-VR closed-loop system is beneficial for customizing the human-avatar mapping based on user's brain activity, which has been limited to users' explicit feedback to date. In order to compensate for the limited signal-to-noise ratio of the BCI output, we incorporated the RL algorithm to identify the optimal magnitude of distortion. The proposed approach successfully

achieved the pseudo-optimal magnitude of distortion and demonstrated its ability to fine-tune levels of assistance for each participant while avoiding degradation of virtual experiences due to BiEs.

Specifically, as shown in previous studies [1, 33, 46] and illustrated in figure 5(a), participants maintained a high level of embodiment even when they could perceive the distortion. In effect, we show that there are three distinct stages of cognitive processing before participants notice and then reject a movement distortion of their avatar. In the first stage, for very low distortions, they are not able to perceive the distortion. In the second stage, between the PoD and the BiE thresholds, participants are still experiencing a strong embodiment for their avatar despite being able to consciously evaluate that the avatar's arm reaching movement is distorted. In the third stage, beyond the BiE threshold, participants reject the distortion as an error of the system, provoking an ErrP that the BCI system can detect. The threshold computed by our BCI-RL approach successfully computed the

pseudo-optimal threshold to be in between the PoD and the BiE thresholds when deploying the personalized ErrP decoder. Importantly, these RL thresholds were nearly identical to those obtained with explicit feedback (1.93 ± 0.23 [mean $\pm$ SE]) [27] (table 1). Thus, thanks to our individually customized distortion magnitude, users performed the reaching task without affecting their experience of embodiment for their avatar.

As opposed to previous works that evaluated the user's brain response at given time points (e.g. after the executions of a movement by an avatar) [19, 20, 47], we performed continuous classification of ErrPs during avatar-based interactions and used the ErrP-BCI decoding outputs to customize the human-avatar mapping. Replacing time-locked by continuous classification of ErrPs is a challenge for BCI due to the uncertainty of EEG signals, but a necessary step for applying it to VR interaction, as users continuously interact with their embodied avatar. Although a previous work showed the customization of robot trajectories through continuous ErrP classification [25], their task did not involve varying magnitudes in erroneous interactions. A recent study [24] revealed the scalability of the ErrP-BCI decoding outputs over the magnitude of errors, and posed the challenge of decoding ErrPs induced by small errors. Critically, our experimental protocol did not include trials with a small magnitude of distortions in the decoder-calibration phase. Nevertheless, our BCI-VR closed-loop system using a personalized decoder successfully classified trials with small distortions and customized the human-avatar mapping, allowing one to maintain embodiment for their avatar.

Furthermore, we also evaluated to what extent the decoder trained with multiple participants generalizes to a new group of participants (non-personalized decoder) [48, 49]. The decoding performance degraded both in time-locked and continuous classification compared to the personalized decoder (figures 4(c) and (d)). The RL threshold obtained with the non-personalized decoder did not reveal significant differences from the PoE nor BiE threshold, while being higher than them. These RL thresholds were also higher than those identified with explicit feedback [27] (table 1). Notably, when the non-personalized decoder was used, participants saved about 90 mins as data collection to calibrate the personalized decoder was omitted. Thus, they were still able to receive implicit BCI feedback immediately after the instruction to customize their human-avatar mapping. Although the non-personalized decoder showed degraded performance (figures 4(c) and (d)), our analysis revealed a progressive increase in the output of the non-personalized decoder output for both time-locked and continuous classification over the magnitude of distortions, but limited to smaller ranges (figures 4(a) and (b)). This suggests the importance of calibrating

the decision threshold when deploying the non-personalized decoder for both time-locked and continuous classification and also the need to update the decoder to increase the range of posterior probabilities. These procedures may further improve precision of RL thresholds when using a non-personalized decoder.

One of the limitations of the present study is the relatively specific human-avatar mapping and BCI algorithms. For example, subjects were instructed to fix their gazes and head movements to preserve EEG signals from possible muscle contamination, and they performed repetitive reaching actions to collect enough data to build a personalized decoder and calibrate their interaction. In an ideal scenario, we expect participants to move freely without repetitive actions, while the BCI-VR closed-loop system implicitly calibrates their human-avatar mapping without the need for collecting data for building a personalized decoder.

In summary, this study demonstrated the possibility and showed the benefits of online adaptation of the human-avatar mapping during VR experiences, without asking explicitly or interrupting the interaction. Although the RL threshold obtained with a non-personalized decoder was not between PoD and BiE thresholds, the decoder may be adaptively updated in an experiment consisting of multiple sessions [24]. Thus, it would be crucially important to test our BCI-VR system over longitudinal sessions because one could speculate that users would increase their sensitivity to perceive visuo-proprioceptive conflicts and become more susceptible to BiE over time. Future studies may include use of the VR system over multiple sessions and evaluate their PoD, BiE and RL threshold, with the possibility to change over sessions while updating the decoder. With increasing involvement of full-body interaction in immersive VR commercial products, BCI-VR systems would be beneficial for optimizing the human-avatar mapping, allowing to maintain an engaging avatar-based interaction and a compelling immersive experience while detecting when users notice a problem and seamlessly correcting it.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: <https://zenodo.org/records/8111105>.

Acknowledgments

This work was supported by the Swiss National Science Foundation (Project 'Immersive Embodied Interactions', 200020.178790), the Hasler Foundation, Switzerland, and by the Swiss National Center of Competence in Research in Robotics (NCCR).

ORCID iDs

Fumiaki Iwane  <https://orcid.org/0000-0002-9659-4127>

Thibault Porssut  <https://orcid.org/0000-0002-6691-1427>

Olaf Blanke  <https://orcid.org/0000-0002-9745-3983>

Ricardo Chavarriaga  <https://orcid.org/0000-0002-8879-2860>

José del R Millán  <https://orcid.org/0000-0001-5819-1522>

Bruno Herbelin  <https://orcid.org/0000-0003-4570-5146>

Ronan Boulic  <https://orcid.org/0000-0001-9176-6877>

References

- Porssut T, Herbelin B and Boulic R 2019 Reconciling being in-control vs. being helped for the execution of complex movements VR *IEEE Conf. on Virtual Reality and 3D User Interfaces VR* pp 529–37
- Kiltner K, Groten R and Slater M 2012 The sense of embodiment in virtual reality *Presence Teleoperators Virtual Environ.* **21** 373–87
- Kokkinara E and Slater M 2014 Measuring the effects through time of the influence of visuomotor and visuotactile synchronous stimulation on a virtual body ownership illusion *Perception* **43** 43–58
- Porssut T, Iwane F, Chavarriaga R, Blanke O, Millán J d R, Boulic R and Herbelin B 2023 EEG signature of breaks in embodiment in VR *PLoS One* **18** 1–13
- Falkenstein M, Hohnsbein J, Hoormann J and Blanke L 1991 Effects of crossmodal divided attention on late ERP components. II. Error processing in choice reaction tasks *Electroencephalogr. Clin. Neurophysiol.* **78** 447–55
- Gehring W J, Goss B, Coles M G H, Meyer D E and Donchin E 1993 A neural system for error detection and compensation *Psychol. Sci.* **4** 385–90
- Holroyd C B and Coles M G H 2002 The neural basis of human error processing: reinforcement learning, dopamine and the error-related negativity *Psychol. Rev.* **109** 679–709
- Pezzetta R, Nicolardi V, Tidoni E and Aglioti S M 2018 Error, rather than its probability, elicits specific electrocortical signatures: a combined EEG-immersive virtual reality study of action observation *J. Neurophysiol.* **120** 1107–18
- Spinelli G, Tieri G, Pavone E F and Aglioti S M 2018 Wronger than wrong: graded mapping of the errors of an avatar in the performance monitoring system of the onlooker *NeuroImage* **167** 1–10
- Pavone E F, Tieri G, Rizza G, Tidoni E, Grisoni L and Aglioti S M 2016 Embodying others in immersive virtual reality: electro-cortical signatures of monitoring the errors in the actions of an avatar seen from a first-person perspective *J. Neurosci.* **36** 268–79
- Raz G, Gurevitch G, Vaknin T, Aazamy A, Gefen I, Grunstein S, Azouri G and Goldway N 2019 Electroencephalographic evidence for the involvement of mirror-neuron and error-monitoring related processes in virtual body ownership *NeuroImage* **207** 116351
- Padrao G, Gonzalez-Franco M, Sanchez-Vives M V, Slater M and Rodriguez-Fornells A 2016 Violating body movement semantics: neural signatures of self-generated and external-generated errors *NeuroImage* **124** 147–56
- Si-Mohammed H, Lopes-Dias C, Duarte M, Argelaguet F, Jeunet C, Casiez G, Müller-Putz G R, Lécuyer A and Scherer R 2020 Detecting system errors in virtual reality using EEG through error-related potentials *IEEE Conf. on Virtual Reality and 3D User Interfaces (VR)* pp 653–61
- Steinhauser R, Wirth R, Kunde W, Janczyk M and Steinhauser M 2018 Common mechanisms in error monitoring and action effect monitoring *Cogn. Affective Behav. Neurosci.* **18** 1159–71
- Rabbitt P A 1966 Errors and error correction in choice-response tasks *J. Exp. Psychol.* **71** 264
- Chang A, Chen C C, Li H H and Li C S R 2014 Event-related potentials for post-error and post-conflict slowing *PLoS One* **9** e99909
- Chavarriaga R and Millán J d R 2010 Learning from EEG error-related potentials in noninvasive brain-computer interfaces *IEEE Trans. Neural Syst. Rehabil. Eng.* **18** 381–8
- Ferrez P and Millán J d R 2008 Error-related EEG potentials generated during simulated brain-computer interaction *IEEE Trans. Biomed. Eng.* **55** 923–9
- Iturrate I, Chavarriaga R, Montesano L, Minguez J and Millán J d R 2015 Teaching brain-machine interfaces as an alternative paradigm to neuroprosthetics control *Sci. Rep.* **5** 13893
- Salazar-Gomez A F, DelPreto J, Gil S, Guenther F H and Rus D 2017 Correcting robot mistakes in real time using EEG signals *IEEE Int. Conf. on Robotics and Automation* pp 6570–7
- Chavarriaga R, Sobolewski A and Millán J d R 2014 Errare machinale est: the use of error-related potentials in brain-machine interfaces *Front. Neurosci.* **8** 208
- Dias C L, Sburlea A I and Müller-Putz G R 2018 Masked and unmasked error-related potentials during continuous control and feedback *J. Neural Eng.* **15** 036031
- Lopes-Dias C, Sburlea A I, Breitegger K, Wyss D, Drescher H, Wildburger R and Müller-Putz G R 2021 Online asynchronous detection of error-related potentials in participants with a spinal cord injury using a generic classifier *J. Neural Eng.* **18** 046022
- Iwane F, Sobolewski A, Chavarriaga R and Millán J d R 2023 EEG error-related potentials encode magnitude of errors and individual perceptual thresholds *iScience* **26** 107524
- Batzianoulis I, Iwane F, Wei S, Correia C G P R, Chavarriaga R, Millán J d R and Billard A 2021 Customizing skills for assistive robotic manipulators, an inverse reinforcement learning approach with error-related potentials *Commun. Biol.* **4** 1406
- Iwane F, Billard A and Millán J d R 2023 Inferring individual evaluation criteria for reaching trajectories with obstacle avoidance from EEG signals *Sci. Rep.* **13** 20163
- Porssut T, Hou Y, Blanke O, Herbelin B and Boulic R 2022 Adapting virtual embodiment through reinforcement learning *IEEE Trans. Vis. Comput. Graph.* **28** 3193–205
- Jauregui D A G, Argelaguet F, Olivier A H, Marchal M, Multon F and Lécuyer A 2014 Toward ‘pseudo-haptic avatars’: modifying the visual animation of self-avatar can simulate the perception of weight lifting *IEEE Trans. Vis. Comput. Graph.* **20** 654–61
- Azmadian M, Hancock M, Benko H, Ofek E and Wilson A D 2016 Haptic retargeting: dynamic repurposing of passive haptics for enhanced virtual reality experiences *Conf. on Human Factors in Computing Systems* pp 1968–79
- Burns E, Razzaque S and Whitton M 2007 MACBETH: management of avatar conflict by employment of a technique hybrid *Int. J. Virtual Real.* **6** 11–20
- Bowman D A and Hodges L F 1997 An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments *Symp. on Interactive 3D Graphics* pp 35–38
- Porssut T, Blanke O, Herbelin B and Boulic R 2022 Reaching articular limits can negatively impact embodiment in virtual reality *PLoS One* **17** e0255554
- Galvan D H, Boulic R, Salomon R, Blanke O and Herbelin B 2018 Self-attribution of distorted reaching movements in immersive virtual reality *Comput. Graph.* **76** 142–52

- [34] Schlögl A, Keinrath C, Zimmermann D, Scherer R, Leeb R and Pfurtscheller G 2007 A fully automated correction method of EOG artifacts in EEG recordings *Clin. Neurophysiol.* **118** 98–104
- [35] Iwane F, Iturrate I, Chavarriaga R and Millán J d R 2021 Invariability of EEG error-related potentials during continuous feedback protocols elicited by erroneous actions at predicted or unpredicted states *J. Neural Eng.* **18** 046044
- [36] Spüler M, Walter A, Rosenstiel W and Bogdan M 2014 Spatial filtering based on canonical correlation analysis for classification of evoked or event-related potentials in EEG data *IEEE Trans. Neural Syst. Rehabil. Eng.* **22** 1097–103
- [37] Iwane F, Chavarriaga R, Iturrate I and Millán J d R 2016 Spatial filters yield stable features for error-related potentials across conditions *IEEE Int. Conf. on Systems, Man and Cybernetics* pp 661–6
- [38] Spüler M 2017 Spatial filtering of EEG as a regression problem *7th Graz Brain-Computer Interface Conf.* (<https://doi.org/10.3217/978-3-85125-533-1-84>)
- [39] Hotelling H 1936 Relations between two sets of variates *Biometrika* **28** 321–77
- [40] Zhang H, Chavarriaga R, Khalilidali Z, Gheorghe L, Iturrate I and Millán J d R 2015 EEG-based decoding of error-related brain activity in a real-world driving task *J. Neural Eng.* **12** 066028
- [41] Sutton R S and Barto A G 1998 *Reinforcement Learning: An Introduction* (MIT Press)
- [42] Lecuyer A, Coquillart S, Kheddar A, Richard P and Coiffet P 2000 Pseudo-haptic feedback: can isometric input devices simulate force feedback? *IEEE Virtual Reality* pp 83–90
- [43] Klein S A 2001 Measuring, estimating and understanding the psychometric function: a commentary *Percept. Psychophys.* **63** 1421–55
- [44] Hajcak G, McDonald N and Simons R F 2003 To err is autonomic: error-related brain potentials, ANS activity and post-error compensatory behavior *Psychophysiology* **40** 895–903
- [45] Gehring W, Liu Y, Orr J and Carp J 2012 The error-related negativity (ERN/Ne) *The Oxford Handbook of Event-Related Potential Components* (Oxford University Press) (<https://doi.org/10.1093/oxfordhb/9780195374148.013.0120>)
- [46] Debarba H G, Perrin S and Boulic R 2018 Perception of redirected pointing precision in immersive virtual reality *IEEE Conf. on Virtual Reality and 3D User Interfaces (VR)*
- [47] Ehrlich S K and Cheng G 2018 Human-agent co-adaptation using error-related potentials *J. Neural Eng.* **15** 066014
- [48] Lopes-Dias C, Sburlea A I and Müller-Putz G R 2020 A generic error-related potential classifier offers a comparable performance to a personalized classifier *42nd Annual Int. Conf. IEEE Engineering in Medicine & Biology Society* pp 2995–8
- [49] Bhattacharyya S, Konar A, Tibarewala D N and Hayashibe M 2017 A generic transferable EEG decoder for online detection of error potential in target selection *Front. Neurosci.* **11** 226

MLOps as Enabler of Trustworthy AI

Yann Billeter, Philipp Denzel,
Ricardo Chavarriaga, Oliver Forster,
and Frank-Peter Schilling
Centre for AI (CAI)
Zurich University of Applied Sciences ZHAW
Winterthur, Switzerland
{bily,denp,char,foro,scik}@zhaw.ch

Stefan Brunner, Carmen Frischknecht-Gruber,
Monika Reif, and Joanna Weng
Institute for Applied Mathematics and Physics (IAMP)
Zurich University of Applied Sciences ZHAW
Winterthur, Switzerland
{brt,frsh,reif,wenj}@zhaw.ch

Abstract—As Artificial Intelligence (AI) systems are becoming ever more capable of performing complex tasks, their prevalence in industry, as well as society, is increasing rapidly. Adoption of AI systems requires humans to trust them, leading to the concept of trustworthy AI which covers principles such as fairness, reliability, explainability, or safety. Implementing AI in a trustworthy way is encouraged by newly developed industry norms and standards, and will soon be enforced by legislation such as the EU AI Act (EU AIA). We argue that Machine Learning Operations (MLOps), a paradigm which covers best practices and tools to develop and maintain AI and Machine Learning (ML) systems in production reliably and efficiently, provides a guide to implementing trustworthiness into the AI development and operation lifecycle. In addition, we present an implementation of a framework based on various MLOps tools which enables verification of trustworthiness principles using the example of a computer vision ML model.

Index Terms—AI, MLOps, explainability, trustworthiness

I. INTRODUCTION

Trustworthy AI (TAI) principles are anticipated to shape future AI system regulations [1]. Despite their crucial role, these principles often remain abstract, lacking concrete operational mandates [2]. While emerging frameworks offer high-level ethical guidelines for TAI, providing high-level addressing principles like fairness, autonomy, control, transparency, reliability, security, and privacy (e.g., [3]), there remains a need for technical guidelines and workflows.

In parallel, yet seemingly unrelated, the new and fast-evolving field of Machine Learning Operations (MLOps) [4] takes inspiration from the concept of DevOps (Development and Operations) to establish methods, best practices, and tools to operationalize an ML- or AI-based system (AIS), i.e., to bring it into production. Covering all stages from project setup and requirements to continuous integration/delivery, data management, model development, testing, validation, deployment (including cloud and edge), monitoring, and continual learning, a systems approach to MLOps ensures holistic alignment with specified objectives [5]. We show that MLOps practices, besides streamlining deployment and maintenance of AIS, can be naturally extended to address and assess specific requirements on AI trustworthiness.

The remainder of this paper is structured as follows: section II reviews existing research on integrating TAI into the ML

lifecycle. In section III, we provide a comprehensive mapping of TAI principles to both the MLOps lifecycle and the practices underlying it. In section IV, we describe a concrete implementation of TAI principles as part of an MLOps system. Finally, section V discusses gaps and obstacles in the adoption of MLOps for TAI and suggests avenues for future research.

II. RELATED WORK

A. Trustworthy AI in the ML lifecycle

Traditional performance metrics in ML need to be complemented with additional principles as real-world applications rise. Technical concerns like robustness, explainability, transparency, reproducibility, and generalization, along with ethical considerations such as fairness, privacy, and accountability collectively fall under the term “trustworthy AI” [6]. This concept is closely linked to “AI governance”, defined as “the set of rules, regulations, ethical and technical frameworks, and similar mechanisms that guide the development and deployment of artificial intelligence technologies.” [7].

Several studies integrate TAI elements into the ML lifecycle. Laato et al. [8] include AI governance in common system development lifecycle models, resembling MLOps. Ashmore et al. [9] outline assurance desiderata for each ML lifecycle stage and review existing methods. Li et al. [6] propose a systematic approach to incorporate TAI in the ML lifecycle. Despite recognizing the importance of a holistic approach to trustworthiness across lifecycle stages, the literature lacks integration of trustworthy AI with the growing adoption of MLOps. Although a guide on adopting MLOps in the context of responsible AI exists [10], it does not directly address how MLOps practices map to TAI principles. Additionally, actual real-world evaluations for existing approaches are scarce.

B. Metrics in MLOps

MLOps is driven by automation and metrics. Quality and reliability, for instance, are measured and monitored through metrics like *mean time to restore* and *change failure rate* (percentage of deployments causing failure in production). In machine learning, metrics guide training and testing. In trustworthy AI, metrics serve a dual purpose. Firstly, they enable the construction of effective feedback loops with quantifiable measures for TAI principles in MLOps. Secondly, they

TABLE I
INTERRELATION OF MLOPS STAGES (COLUMNS), MLOPS PRACTICES (ROWS), AND TAI PRINCIPLES, I.E., AUTONOMY AND CONTROL (AC), FAIRNESS (FN), PRIVACY (PR), RELIABILITY (RL), SECURITY (SE), TRANSPARENCY (TR). A “✓” DENOTES MLOPS PRACTICES APPLICABLE TO RESPECTIVE STAGES. TAI PRINCIPLES LISTED AFTER THE “✓” INDICATE A PARTICULAR RELEVANCE OF THE PRACTICE TO THE LISTED PRINCIPLES.

Practice / Stage	Business & Data Understanding	Data Engineering	ML Model Engineering	ML Model Evaluation	Deployment	Monitoring & Maintenance
TAI Principles	FN, PR, RL	FN, PR, TR, RL	AC, FN, PR, TR, RL	FN, RL, TR	AC, PR, RL, SE, TR	AC, FN, RL, SE
Versioning		✓: TR, RL	✓: TR, RL		✓	
Testing		✓: FN, TR, RL		✓: FN, RL, TR		
Automation		✓	✓	✓	✓	
Reproducibility		✓	✓	✓	✓	
Deployment					✓	
Monitoring						✓

may serve in definitions of and compliance to (upcoming) AI regulation. An example in the context of the EU AIA is the Key AI Risk Indicators framework for AI in the financial services industry [11]. Yet, despite their practical relevance and expected importance, the role of metrics in TAI has not been fully addressed in the existing literature.

III. MLOPS AS ENABLER OF TRUSTWORTHY AI

In this section, we illustrate how MLOps enables TAI by establishing the relationship of trustworthy AI principles with MLOps concepts. For these principles, we rely on the six dimensions of trustworthiness identified by the Fraunhofer Institute [3]. For each principle, we report (a) the key stage(s) of the ML lifecycle where it should be addressed, as well as how it relates to the MLOps practices [12] (versioning, testing, automation, reproducibility, deployment, and monitoring), and (b) which metrics could be tracked to monitor progress of the respective TAI principle. We follow the CRSIP-ML(Q) lifecycle [13] for our MLOps lifecycle model, comprising the stages “Business & Data Understanding”, “Data Engineering”, “ML Model Engineering”, “ML Model Evaluation”, “Deployment”, and “Monitoring & Maintenance”. Table I summarizes the relation of trustworthy AI principles, MLOps lifecycle stages, and MLOps practices.

A. Mapping Trustworthy AI Principles to the MLOps Lifecycle

a) *Autonomy and Control (AC)*: This principle focuses on two key elements: evaluating the appropriate level of autonomy for the artificial intelligence application (e.g., human-in/on/out-of-the-loop) and analysing how well the AI application supports and allows adequate room for the individual’s interaction with it. The key stages in the ML lifecycle for autonomy and control are model engineering, deployment, and monitoring & maintenance. Model engineering considers potential human feedback. Deployment influences human control and model use. Monitoring is vital for human oversight in the maintenance stage. Despite the importance of addressing these concerns, there is limited research on achieving optimal human control in the context of AI systems, and we are unaware of potential metrics for this principle.

b) *Fairness (FN)*: The fairness principle mainly aims to prevent unjust discrimination in AI use, often caused by biased training data or statistical under-representation of certain groups, leading to reduced quality for those groups. Regulatory requirements highlight the importance of addressing fairness in the planning stage. Technical interventions in data engineering, model engineering, and evaluation can address issues like class imbalances and group representation. Constraint enforcement and ongoing monitoring in the model engineering and monitoring stages, respectively, ensures compliance with fairness criteria, with metrics categorized into individual, group, and causality-based fairness [14].

c) *Privacy (PR)*: This principle focuses on safeguarding sensitive data during AIS development and operation, encompassing personal data and business secrets. Similar to fairness, privacy requirements often stem from regulations, necessitating early consideration in the planning stage, such as through a privacy-by-design approach. Technical interventions in data and model engineering involve practices like data minimization, reducing attack surfaces, and implementing differential privacy (DP) [15]. Privacy-related issues, including model extraction and attacks like membership inference and model inversion [16], must be addressed during deployment. Privacy metrics serve a dual purpose: information-theoretic measures quantify system privacy properties, while in the context of DP, metrics like summary statistics assess the inherent privacy-utility tradeoff [17], [18].

d) *Reliability (RL)*: The reliability principle assesses the AIS’ quality, focusing on robustness and output uncertainties. Emphasized during model engineering and evaluation, practices like certified training can enhance robustness [19]. Evaluation includes testing model robustness via adversarial attacks and formal verification [20], with certified accuracy as a popular metric. Reliability extends to deployment, addressing potential user adversarial actions and ensuring fast recovery. Monitoring and maintenance involve close attention to reliability metrics. An emerging approach involves integrating domain knowledge [21], improving reliability and aiding in judging prediction plausibility. Identifying applicable domain knowledge should begin in the planning stage, becoming actionable in the

model engineering and evaluation stages.

e) Security (SE): Involving functional security properties and safeguarding against attacks, Security encompasses measures related to embedding the AI component. This includes traditional IT security methods and metrics (e.g. [22]). It directly correlates with deployment and monitoring stages, akin to traditional software solutions. Beyond established practices for securing AIS, it must address the potential of AI to enhance or compromise existing security measures [23].

f) Transparency (TR): Transparency in an AIS involves interpretability, reproducibility, and explainability, assessing user and expert comprehensibility, as well as result reproducibility and explainability. Improving transparency aligns with MLOps practices such as reproducibility and versioning, which are crucial for comprehensive change tracking. The incorporation of explainable algorithms and architectures enhances model transparency. Throughout model evaluation, monitoring, and maintenance, transparency is assessed using explainability methods [24], providing insights for developers and potential users. However, the absence of a standardized transparency measure requires further research, as machine-based metrics may not directly align with human relevance [25].

B. Trustworthy AI as an Iterative Process in MLOps

Due to the evolving nature of AIS and their complex post-deployment, trustworthiness levels may fluctuate dynamically. Continuous risk monitoring is crucial for responsible AI development, aligning with the iterative nature of MLOps driven by versioning, automation, testing, deployment, and monitoring. The core tenet of continuous refinement can be extended by the addition of trustworthiness metrics alongside accuracy and efficiency. This enables the implementation of continuous feedback loops, systematically addressing TAI requirements throughout the entire AI lifecycle.

IV. CASE STUDY

In the preceding section, we presented how TAI principles map onto the MLOps practices and lifecycle stages. This section shows an implementation of MLOps-supported pipeline for the development of Trustworthy AI systems. Using open-source MLOps software components, we automate the assessment of reliability and transparency requirements for computer vision (CV) models across the data engineering, model engineering, and model evaluation stages. We evaluate the system's capability to monitor trustworthiness for various CV models and tasks, employing various reliability and transparency methods [26]. For instance, for reliability, robustness to perturbations is expected, which we evaluate on both the data and model level, e.g. through estimation of noise profiles of the training data, and simulation of domain-specific perturbations. Regarding transparency, one example is the identification of global explanations to model decisions as part of model evaluation. For this, we extract factors of variation and prototypes during the data engineering stage. During model evaluation, the prototypes are used conjointly with explanation methods [24].

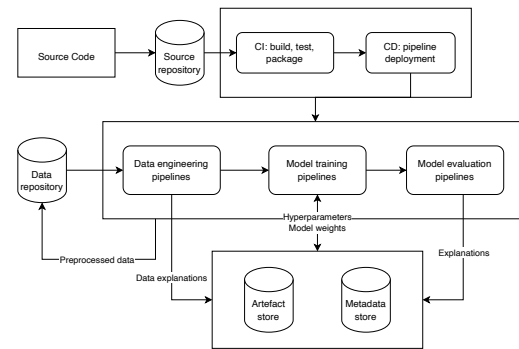


Fig. 1. Overview of the MLOps system architecture.

A. MLOps System Architecture

A spectrum of MLOps tools, encompassing both open-source and proprietary solutions, is available. In alignment with considerations for intellectual property and data protection, we opted for established open-source tools suitable for in-house deployment, except for the code repository, where we utilize git versioning through GitHub¹. We employ Oxen² for data versioning, while MLflow³ manages experiment and model version tracking. Our system integrates GitHub Actions for CI/CD and relies on Apache Airflow⁴ for efficient workload scheduling. A schematic overview is given in fig. 1.

B. Workflow

The system reacts to changes in source code or data, initiating CI/CD processes for code changes and triggering related pipelines. Model code changes lead to the execution of training and explanation pipelines. All resulting artifacts (e.g., model weights, metrics) are stored and versioned in MLflow for analysis.

Data changes trigger data engineering pipelines, generating training-ready data and artifacts (e.g. prototypes and noisy samples) for model evaluation pipelines. This process includes model retraining and evaluation, in turn generating model explanations and reliability diagrams. The automated reliability and transparency feedback loops prevent inadequate models from reaching production. Monitoring data variation factors and noise profiles aids in understanding cause-effect relations between data changes and model performance, guiding future data collection decisions. Ultimately, consistent adherence to these MLOps feedback loops leads to a continuous and automatic increase in trustworthiness.

V. DISCUSSION AND CONCLUSION

MLOps systems ought to be non-negotiable for organizational productivity when developing or deploying AIS. Beyond productivity, they enable TAI through automated feedback loops and metrics, ensuring continuous assessment of TAI principles

¹ <https://github.com/> ² <https://www.oxen.ai/> ³ <https://mlflow.org/>

⁴ <https://airflow.apache.org/>

across lifecycle stages. Using an appropriately scaling MLOps system, this can be achieved regardless of project scale.

The framework also supports assessing conformity with internal or regulatory criteria. The relationship of MLOps and TAI evolves bidirectionally: Akin to how privacy-by-design was developed from a specific concern, MLOps evolves according to such criteria. Conversely, upcoming standards on TAI are informed by the technological state-of-the-art, which entails MLOps.

The described system for MLOps-guided TAI-compatible ML development is not exhaustive; real-world applications necessitate appropriate organizational structures and practices, and adopting MLOps for TAI presents challenges, warranting further research into specific hindrances:

Firstly, significant gaps exist among different trustworthy AI principles, both in research and practical implementation. Recent attention has focused on robustness and security, while autonomy and control remains largely unexplored in the trustworthiness literature. Secondly, some principles are missing adequate metrics. Transparency and explainability methods mainly rely on human interpretation and thus cannot be monitored automatically as part of an MLOps feedback loop, or are machine-generated yet not guaranteed to be relevant for human comprehension.

Finally, the overall implementation of MLOps is often impeded by a lack of clear standardization among MLOps tools. This compels practitioners to develop custom interfacing code for integrating their data and models, increasing friction and impeding the widespread adoption of MLOps tools.

MLOps has the potential to enable trustworthy AI by creating continuous feedback loops throughout the entire AI lifecycle. To leverage this potential, we believe that future research should address reducing the friction encountered in MLOps adoption and seek to address the gaps in the existing trustworthy AI literature.

ACKNOWLEDGMENT

This work was co-financed by Innosuisse (101.650 IP-ICT).

REFERENCES

- [1] EU Parliament. EU AI Act: first regulation on artificial intelligence. [Online]. Available: <https://www.europarl.europa.eu/news/en/headlines/society/20230601STO93804/eu-ai-act-first-regulation-on-artificial-intelligence>
- [2] L. Lucaj, P. van der Smagt, and D. Benbouzid, "AI Regulation Is (not) All You Need," in *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*, ser. FAccT '23. ACM, 2023, p. 1267–1279.
- [3] M. Poretschkin *et al.*, "Leitfaden zur Gestaltung vertrauenswürdiger Künstlicher Intelligenz," Fraunhofer-Institut für Intelligente Analyse- und Informationssysteme IAIS, Technical Report, 2021. [Online]. Available: <https://www.iais.fraunhofer.de/de/forschung/kuenstliche-intelligenz/ki-pruefkatagol.html>
- [4] D. Kreuzberger, N. Kühl, and S. Hirschl, "Machine Learning Operations (MLOps): Overview, Definition, and Architecture," *IEEE Access*, vol. 11, pp. 31 866–31 879, 2023.
- [5] C. Huyen, *Designing Machine Learning Systems*. USA: O'Reilly Media, 2022.
- [6] B. Li, P. Qi, B. Liu, S. Di, J. Liu, J. Pei, J. Yi, and B. Zhou, "Trustworthy AI: From Principles to Practices," *ACM Comput. Surv.*, p. 1–46, 2023. [Online]. Available: <https://doi.org/10.1145/3555803>
- [7] Carnegie Council for Ethics in International Affairs. AI governance. [Online]. Available: <https://www.carnegiecouncil.org/explore-engage/key-terms/ai-governance>
- [8] S. Laato, T. Birkstedt, M. Mäntymäki, M. Minkkinen, and T. Mikkonen, "AI governance in the system development life cycle: insights on responsible machine learning engineering," in *Proceedings of the 1st International Conference on AI Engineering: Software Engineering for AI*, ser. CAIN '22. ACM, May 2022.
- [9] R. Ashmore, R. Calinescu, and C. Paterson, "Assuring the Machine Learning Lifecycle: Desiderata, Methods, and Challenges," *ACM Computing Surveys*, vol. 54, no. 5, pp. 1–39, May 2021.
- [10] B. M. A. Matsui and D. H. Goya, "MLOps: a guide to its adoption in the context of responsible AI," in *Proceedings of the 1st Workshop on Software Engineering for Responsible AI*, ser. ICSE '22. ACM, May 2022, p. 45–49.
- [11] P. Giudici, M. Centurelli, and S. Turchetta, "Measuring ai safety," *SSRN Electronic Journal*, 2022. [Online]. Available: <http://dx.doi.org/10.2139/ssrn.4298352>
- [12] L. Visengeriyeva, A. Kammer, I. Bär, A. Knies, and M. Plöd. (2020) MLOps Principles. [Online]. Available: <https://ml-ops.org/content/mlops-principles>
- [13] S. Studer, T. B. Bui, C. Drescher, A. Hanuschkin, L. Winkler, S. Peters, and K.-R. Müller, "Towards CRISP-ML(Q): A Machine Learning Process Model with Quality Assurance Methodology," *Machine Learning and Knowledge Extraction*, vol. 3, no. 2, pp. 392–413, 2021. [Online]. Available: <https://www.mdpi.com/2504-4990/3/2/20>
- [14] A. Castelnovo, R. Crupi, G. Greco, D. Regoli, I. G. Penco, and A. C. Cosentini, "A clarification of the nuances in the fairness metrics landscape," *Scientific Reports*, vol. 12, no. 1, p. 4209, 2022.
- [15] A. E. Ouadrhiri and A. Abdelhadi, "Differential Privacy for Deep and Federated Learning: A Survey," *IEEE Access*, vol. 10, pp. 22 359–22 380, 2022.
- [16] S. V. Dibbo, "SoK: Model Inversion Attack Landscape: Taxonomy, Challenges, and Future Roadmap," in *2023 IEEE 36th Computer Security Foundations Symposium (CSF)*. IEEE, 2023, pp. 439–456.
- [17] M. Bloch, O. Gunlu, A. Yener, F. Oggier, H. V. Poor, L. Sankar, and R. F. Schaefer, "An Overview of Information-Theoretic Security and Privacy: Metrics, Limits and Applications," *IEEE Journal on Selected Areas in Information Theory*, vol. 2, no. 1, pp. 5–22, 2021.
- [18] J. P. Near and D. Darais, "Guidelines for Evaluating Differential Privacy Guarantees," National Institute of Standards and Technology, Tech. Rep., 2023. [Online]. Available: <https://csrc.nist.gov/pubs/sp/800/226/fpd>
- [19] N. Jovanović, M. Balunovic, M. Baader, and M. T. Vechev, "On the Paradox of Certified Training," *Trans. Mach. Learn. Res.*, vol. 2022, 2021. [Online]. Available: <https://api.semanticscholar.org/CorpusID:252873705>
- [20] L. Li, T. Xie, and B. Li, "SoK: Certified Robustness for Deep Neural Networks," in *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2023, pp. 1289–1310.
- [21] N. M. Gürel, X. Qi, L. Rimanic, C. Zhang, and B. Li, "Knowledge enhanced machine learning pipeline against diverse adversarial attacks," in *Proceedings of the 38th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, M. Meila and T. Zhang, Eds., vol. 139. PMLR, 18–24 Jul 2021, pp. 3976–3987. [Online]. Available: <https://proceedings.mlr.press/v139/gurel21a.html>
- [22] Y. Cheng, J. Deng, J. Li, S. A. DeLoach, A. Singhal, and X. Ou, *Metrics of Security*. Springer International Publishing, 2014, pp. 263–295.
- [23] I. H. Sarker, M. H. Furhad, and R. Nowrozy, "AI-Driven Cybersecurity: An Overview, Security Intelligence Modeling and Research Directions," vol. 2, no. 3.
- [24] C. Molnar, *Interpretable Machine Learning*, 2nd ed., 2022. [Online]. Available: <https://christophm.github.io/interpretable-ml-book>
- [25] F. Biessmann and D. Refiano, "Quality Metrics for Transparent Machine Learning With and Without Humans In the Loop Are Not Correlated," in *Workshop on Theoretical Foundations, Criticism, and Application Trends of Explainable AI*, 2021. [Online]. Available: <https://arxiv.org/abs/2107.02033>
- [26] P. Denzel, S. Brunner, P.-P. Luley, C. Frischknecht-Gruber, M. U. Reif, F.-P. Schilling, A. Amini, M. Repetto, A. Iranfar, J. Weng, and R. Chavarriaga, "A framework for assessing and certifying explainability of health-oriented ai systems," nov 2023, explainable AI in Medicine Workshop, Lugano, Switzerland, 2-3 November 2023. [Online]. Available: <https://digitalcollection.zhaw.ch/handle/11475/29258>

Towards the Certification of AI-based Systems

Philipp Denzel*, Stefan Brunner*, Yann Billeter,
Oliver Forster, Carmen Frischknecht-Gruber, Monika Reif,
Frank-Peter Schilling, Joanna Weng, and Ricardo Chavarriaga

* Co-first author.

Zurich University of Applied Sciences ZHAW

Winterthur, Switzerland

{denp,brtt,bily,foro,frsh,reif,scik,wenj,char}@zhaw.ch

Amin Amini, Marco Repetto, and Arman Iranfar
CertX AG

Fribourg, Switzerland

{amin.amini,marco.repetto,arman.iranfar}@certx.com

Abstract—Certifying the trustworthiness of Artificial Intelligence (AI)-based systems based on dimensions including reliability and transparency is crucial given their increased uptake. Likewise, as regulatory requirements are established, actionable guidelines for certification will be useful for developers and certification bodies to ensure trustworthiness of AI. Here, we present an ongoing effort to develop a validated AI certification scheme which is a framework for assessing the trustworthiness of AI systems including specific objectives with their corresponding means of compliance (i.e. process, documentation or technical methods). Importantly, the scheme makes an explicit link between legal requirements and validated techniques for assessing the compliance of AI systems, resulting in the implementation of a workflow to support AI certification. We explain the rationale for developing the certification scheme and demonstrate the assessment of an example use case with a concrete workflow traversing from objectives to corresponding means, focused on reliability and transparency.

Index Terms—Artificial Intelligence, Machine Learning, Certification, Reliability, Transparency

I. INTRODUCTION

Artificial intelligence (AI) and machine learning (ML) are transforming numerous domains, including those with significant implications for human safety, asset protection, and environmental impact. Ensuring the safe and reliable use of AI technology is therefore of utmost importance. Although there is a large number of ethical guidelines for AI [1], their translation into specific methods and practices remains poorly defined. We address this gap by proposing a certification scheme for AI systems. This scheme consists of two parts:

- (i) a framework outlining existing regulatory requirements, criteria, and measures needed to certify an AI/ML system;
- (ii) guidance linking these requirements to a set of technical and scientific methods for assessing certification-relevant characteristics of AI/ML systems including, among others, reliability and transparency.

Altogether, the certification scheme provides a comprehensive workflow to identify and apply methods and processes for assessing compliance with emerging AI regulations. This work is based on existing efforts on the standardization and regulation of AI systems, and on initial groundwork for AI certification processes. Complementarily, it also draws on recent work in the AI/ML research community on algorithmic methods for determining and verifying relevant properties of ML models.

In the following sections, we provide an overview of existing norms, standards (Section II), and related work (Section III). We then describe our approach for developing the certification scheme (Section IV), and give a brief summary of existing technical methods for assessing aspects of transparency and reliability of AI systems (Section V), complemented by an example of their implementation within a concrete assessment workflow (Section VI).

II. OVERVIEW OF AI LEGISLATION AND STANDARDS

There is a global push for regulatory frameworks for AI. The European Union (EU) has taken a leading role in this, launching efforts for establishing a comprehensive regulatory framework for AI systems in the EU (AI Act) [2]. In fact, as of December 2023, the European Parliament and Council agreed upon a first draft of a legislative proposal [3].

The proposed legislation classifies AI systems into four risk categories based on their potential harm to individuals, society, and the environment: *unacceptable risk*, *high risk*, *limited risk*, and *minimal risk*. AI practices with unacceptable risk, in particular biometric classification, emotion recognition in the workplace, subliminal manipulation to circumvent free will, and social scoring will be prohibited by the legislation, while high-risk AI systems will be subject to additional obligations, such as an assessment of the impact on fundamental rights, especially with regard to the environment and marginalized groups. Moreover, certain AI systems intended for use in high-risk sectors, healthcare, transport, and energy among others, must undergo a conformity assessment process and be certified by a designated third-party assessment body before they can be placed on the EU market. The certification process will assess the AI system's compliance with mandatory requirements such as transparency, accuracy, and robustness, as well as the potential risks associated with its use. For limited-risk AI applications there will be an essential obligation to be transparent about their purpose, function, and decision-making processes. Similar regulatory frameworks are expected to be introduced in other regions of the world. Indeed, in October 2023, the US followed suit by issuing an executive order on the development of new standards for safe, secure, and trustworthy AI [4].

Standards support binding laws and regulations by documenting the state of the art and best practices, and providing a basis for demonstrating compliance and certification. Several organizations and initiatives such as ISO/IEC, IEEE and NIST are currently working on developing relevant AI standards. The standards developed by ISO/IEC [5] cover a wide range of AI aspects, including terminology, performance metrics, data quality, ethics, and human-AI interaction. The IEEE P7000 series of standards [6] focuses on the ethical implications of AI technologies, while the NIST framework [7] provides guidance on managing risks, ensuring data quality, and promoting transparency and accountability in AI systems.

III. RELATED WORK

Various national and international organizations are working on initiatives to support the certification of AI systems. DIN/DKE provide comprehensive recommendations for standardization across all AI topics to establish a common language, principles for development, use, and certification [8]. In this context, the Fraunhofer Institute has developed a guideline for assessing the trustworthiness of AI [9] which proposes to base such evaluations on six dimensions: fairness, autonomy & control, transparency, reliability, safety & security, and privacy. The LNE's AI certification programme provides objective criteria for the selection of trustworthy AI systems based on their evaluation according to compliance with a set of criteria related to ethics, safety, transparency and privacy [10]. Similarly, the IEEE has established a certification programme to assess the transparency, accountability, bias and privacy of AI-related processes [11]. While these initiatives provide detailed description of processes, they remain rather vague when it comes to the evaluation of AI systems with concrete technical methods.

EASA is the first European agency to publish guidance on the safe use of ML [12]. Its purpose is to assist aviation stakeholders in the development and implementation of ML systems with low levels of automation and covers the entire life cycle, including topics such as the development process, data collection, model selection and testing, and the use of ML operations (MLOps). As such, it serves as one of the main precursors to the development of a comprehensive AI certification scheme outlined in this contribution.

Companies such as IBM, Meta, Seldon, Microsoft, and Google offer open source toolboxes for AI evaluation such as [13]–[20]. These toolboxes aim to detect and mitigate bias, explain model decisions, ensure the robustness, and assess the uncertainty of AI systems.

IV. TOWARDS A CERTIFICATION SCHEME

Our goal is to design an actionable guide for regulators and developers to be able to assess and certify the trustworthiness of AI-based products and their associated processes, including data, development, model, testing, and (continuous) operation. This requires considering all stakeholders, including users, developers, auditors, and authorities, as well as the entire life

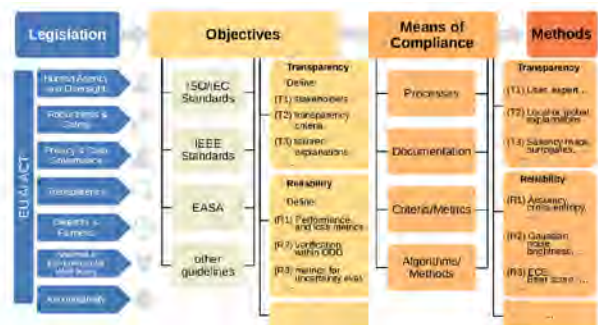


Fig. 1: Certification scheme for AI systems. It enables identification of application-specific requirements, respective means of compliance and technical methods for conformity assessment (see text).

cycle of the AI-system from its conception to its deployment, and in principle even its decommissioning.

While numerous guidelines for the establishment of responsible AI exist (see Section III), most lack detailed prescriptions for actual measures to guarantee trustworthiness. The majority of these guidelines base the definition of trustworthiness on the same ethical principles which only differ in the importance with which they weight each dimension. Our research provides clear guidance on means for assessing the following trustworthiness dimensions:

- *Reliability* ensures that the AI system can operate effectively under various conditions and resist (input) errors, bias or malicious attacks.
- *Transparency* ensures that different stakeholders can understand the decision-making process of an AI-system and thus can judge the extent to which they can trust it.
- *Safety* is crucial for AI systems in critical applications like healthcare or self driving vehicles to prevent any harm or unintended consequences.
- *Autonomy and control* is related to the autonomy of the AI system (“Human-in/on-the-Loop” vs “Complete Autonomy” [21]) as well as the level of support of human decision making.

For brevity, in this paper we focus in the following on the reliability and transparency dimensions.

The certification scheme is illustrated in Fig. 1. The scheme identifies application-specific *objectives* for achieving conformity with the requirements of EU legislation. Then, the associated *means of compliance* are identified starting from the guidelines summarized in Section III and taking also the state of the art into account. Currently, the certification scheme comprises more than 230 *objectives* for certification and about 250 *means of compliance*, based on the analysis of 38 documents on regulations and standards.

Corresponding processes, algorithms, and technical *methods* (see next section) are identified in the next step, which were tested and validated to assess the certification objectives. The choice of appropriate methods can vary significantly based on several factors, including the data type, model type, the phase

of the life cycle, number of components in the AI system, and the stakeholders involved. The certification scheme incorporates all of these factors and provides a clear guidance on how to proceed.

To confirm the suitability of the technical and algorithmic methods for not just an academic setting, but for certification of real-world applications, we validate the methodology on use cases featuring actual AI applications provided by industrial partners¹. This validation process ensures the certification scheme is comprehensive, reliable, and effective.

Among the means of compliance for the existing regulatory objectives, there are in particular:

- use of appropriate quantifiable metrics or qualitative criteria refining the objective;
- documentation of system design, implementation, validation, deployment and monitoring;
- adherence to processes defined by best practices and guidelines;
- use of technical and algorithmic methods for verification and testing of certain properties of the AI system.

V. TECHNICAL METHODS FOR AI CERTIFICATION

The term technical and algorithmic method refers to methods discussed by the AI scientific literature, open-source code implementing these algorithms (e.g. code repositories accompanying publications, or reference implementations by third parties), or commercial and non-commercial software solutions for verification and testing of AI systems. The AI research community is actively engaged in proposing and advancing such technical and algorithmic methods to assess the behavior, robustness, interpretability, and overall trustworthiness of AI systems.

In particular, research in methods to assess AI systems along the transparency and reliability dimensions is rapidly advancing. Methods for transparency generally assess the behavior of the model with respect to the interpretability of its input and output, as well as the human comprehensibility of the model's internal mechanisms leading to predictions [22]. Regarding reliability, areas of concern include model accuracy and uncertainty, dataset coverage or bias, model generalization properties, and robustness with respect to noise, labeling errors or adversarial attacks [23].

Despite considerable efforts from the communities in these fields, many problems still persist for the certification of AI systems. Since most systems requiring certification are deployed in real-world environments, the suitability of the technical methods is barely known. Mostly due to the lack of realistic benchmarks, new algorithms are often tested in an academic setting against idealized datasets.

A. Methods for Transparency

ML techniques like Deep Learning (DL) scale well with the amount of data which feed their learning process. This scalability often comes at the cost of increased model complexity,

which in turn makes the rationale behind the models' predictions opaque to users and experts alike. Yet, certification will include objectives for model interpretability and explainability, related to the capacity of providing explanations for different stakeholders, e.g., users, affected citizens, domain experts, or government regulators².

Moreover, different types of models and data modalities potentially demand different methods or variations thereof.

Contrary to intrinsically interpretable models such as decision trees or linear regression, models (e.g. DL models) whose internal behavior and predictions cannot be understood by looking at their parameters, are considered *black-box models* in this context. Consequently, they require additional methods to gain insight into their decision-making process.

These techniques for instance analyze the dependence of the model's output on the data feature statistics, visualize model internals, or find *surrogate* models, interpretable models which approximate the model's behavior *locally*, on single data samples, or *globally*, on data features in general and the entirety of the data domain.

A comprehensive taxonomy of well-established explainability methods provides organizing principles to identify appropriate methods for a given problem. E.g., [22], [24] propose such a taxonomy encompassing many well-established methods in ML research. Both schemes are presented in the form of decision trees and essentially only differ in the sequence of their questions, thus we list them here in arbitrary order:

- Data or model?* Does the explanation pertain to the data or the model? Data explanations can take form as samples [25], features [26], or distributions.
- Intrinsic or post-hoc?* Intrinsically interpretable models usually have no need for additional explanations, whereas black-box models involve auxiliary methods employed after training.
- Local or global?* Does the explanation need to describe the behavior of the entire model (global) or an individual prediction (local)? Can the explanations take into account a certain class of model? E.g., methods which work only for neural networks are model-specific, whereas model-agnostic methods assume a black-box model and analyze the relation between input and output without access to the model's internal structure.
- Interpretation type?* Some cases require a specific form of explanation such as feature summary statistics [27], saliency or attribution maps [28]–[30], visualizations of features or model internals [31], exemplar explanations, so-called prototypes [25], or explanations through surrogates [32].

These taxonomies facilitate the classification and search for transparency methods that meet the requirements of the certification scheme. Selected methods, in combination with explainability metrics aid in the identification of explainability

¹to secure their anonymity upon request, the use case presented in this paper is completely unrelated to these test cases.

²Note that in the literature some make a clear distinction between the terms interpretable and explainable; here we use the terms interchangeably

gaps for users and developers [33]; many are implemented in open-source toolkits, e.g. [16]–[18]. The research field of explainable AI is rapidly advancing [34], [35]. However, research in corresponding metrics is still lacking.

B. Methods for Reliability

In the context of reliability, it is essential to encompass the entire intended operational domain to ensure that an AI system will behave reliably under all possible circumstances that it may encounter. This scope can be effectively described by the Operational Design Domain (ODD), a concept from the automotive sector as defined by SAE International [36], which we extend here to other AI applications. The ODD delineates the specific conditions under which an AI system is designed to operate safely, robustly and accurately, including inherent or environmental disturbances such as electronic noise in camera sensors, lens flare in image or video processing scenarios, and environmental conditions such as illumination.

A fully covered ODD is prerequisite for the verification of the AI system's reliability. The examination of its ability to perform accurately, robustly, and in difficult situations is, however, heavily dependent on the derivation of meaningful tests. For this purpose, the proposed scheme provides a set of methods for assessing performance, robustness, and uncertainty. In this context, reliability relates to the consistent and accurate performance across various scenarios. *Robustness* encapsulates the system's ability to resist against perturbations, be they intentional attacks [23], such as (synthetic) adversarial examples, inherent variations in the input data, or environmental perturbations [37]. Precise *uncertainty* quantification is another critical aspect for assessing reliability of an AI system, estimating the model's limitation to encompass the deep, intrinsic complexity and dynamics of real-world environments.

For assessing performance and robustness of AI systems, confidence interval (CI) tests [38] can be conducted. They evaluate the model based on test data featuring different types of noise simulations over a range of amplitudes. Alternatively, abstract interpretation as in [39], [40] also tests robustness with the advantage that these methods need significantly less computing time compared to the CI tests.

While there are numerous metrics for evaluating performance or robustness (e.g. accuracy, recall, mAP), where the choice depends strongly on the application, the metrics specifically for uncertainty estimation are not as widely established. In the context of the certification scheme, we will focus on the following selected metrics:

- *Calibration* is measured by the so-called reliability diagram [41], [42]. The diagram plots the accuracy as a function of model confidence for different intervals. Here, accuracy measures the mean of the correct predictions for several interval thresholds, while confidence measures the mean of the predicted probabilities for these thresholds. As a metric, it assesses whether the AI system has the same confidence level as the prediction accuracy.
- *Expected Calibration Error (ECE)* [43]. It derives the average between the absolute difference of the accuracy

and confidence for an ML model for different thresholds bins. An ECE score close to zero implies a correct model calibration, while larger values indicate a discrepancy.

- *(Multi-class) Brier score* [44] is another metric for the assessment of the model certainty which measures the accuracy of probabilistic predictions of the AI system, with a score of zero indicating optimal model calibration.

Alternatively, the intrinsic and extrinsic uncertainty of AI systems can be assessed using Bayesian neural networks or dropout at test-time [45] in combination with Monte-Carlo sampling and uncertainty decomposition [46].

C. Putting Certification into Practice

The assessment of the means of compliance requires suitable and efficient selection and implementation of the selected methods as described in Section IV. In the context of ML, the paradigm of MLOps aligns nicely with this approach and provides effective tools towards an implementation of such a scheme [47]. An MLOps pipeline encompassing both ML methods and data enables complete experiment tracking, and contributes to the reproducibility and auditability of the AI product. Such a pipeline can be designed with various degrees of automation. Regulatory requirements, in particular for high-risk applications, have identified pipeline orchestration, model tracking, including versioning, and data versioning as imperative key components.

While certainly not the only choices, we are addressing these requirements using the following tools:

- Orchestration of training and testing is carried out through GitHub Actions and Airflow (<https://airflow.apache.org/>).
- MLflow provides an analytics framework, giving insights into the (hyper)parameters of the model and the method at hand (<https://mlflow.org/>).
- Oxen was chosen for data version control and dataset management (<https://www.oxen.ai/>).

Note that there are complete, cloud-based services providing a similar feature set as the aforementioned pipeline. However, for many companies, privacy concerns prohibit the use of external facilities due to the potential loss of data ownership. The described pipeline on the other hand is containerized and can easily run on any cluster or workstation, enabling an in-house assessment.

VI. PRACTICAL EXAMPLE

We illustrate the workflow of the certification scheme on a small subset of objectives with a practical example of a system for classifying dermatoscopic images of skin lesions using a convolutional neural network (CNN) model.

Regulatory requirements for systems performing medical diagnoses including an AI component will demand the definition of an appropriate degree of transparency and reliability. As a full assessment would be too extensive to be included here, we report only a few exemplary requirements for each dimension, see Table I.

TABLE I: Three examples for transparency and reliability objectives specified in the certification scheme, each applied to the skin lesion classification use case.

	Objective	Means of compliance	Practical example
Transparency	(T1) Define stakeholders.	Consider all relevant stakeholders. Document the choice.	Physicians, developer.
	(T2) Define necessary transparency criteria.	Document the criteria to be used for evaluating the transparency: - Scope, design, and degree of transparency of the procedures. - Depth and breadth of introspection considering the model outputs.	Explanations: - Local (physician, developer) - Global (developer)
	(T3) Define suitable transparency methods.	Define suitable set of transparency methods. Justify selection.	SHAP, CAM
	...		
Reliability	(R1) Define performance and loss metrics.	Define suitable set of performance and loss metrics. Justify selection.	Accuracy, Cross-Entropy Loss
	(R2) Perform and document verification of the trained model within the ODD.	Execute tests on ideal data. Execute tests including test cases covering: - Perturbations due to fluctuations in the input (e.g. noise on sensors). - Edge cases that can arise on the data within the ODD (e.g. light). - Combined effects.	Perturbations: - Gaussian noise - Brightness - Brightness + Gaussian noise
	(R3) Define metrics for uncertainty evaluation.	Define suitable set of metrics. Justify selection.	ECE, Brier score
	...		

For any use case, the data modality, model type, and its task dictate to a large extent the appropriate means of compliance for its objectives. Thus, it is important that rationales and justifications for the choice of training data and model architectures are properly documented.

For this application we used the ISIC2019 dataset [48], which consists of 25,331 dermoscopic images from nine diagnostic categories, e.g. *melanoma* and *basal cell carcinoma*, *vascular lesions*. These images show variations in lesion characteristics such as asymmetry, margin, color and diameter, and present challenges such as class imbalance, unknown test set classes, and different image resolutions and imaging protocols.

For the AI component we used the small EfficientNetV2 (EfficientNetV2B0) architecture [49] as feature extraction backbone and extended it by a fully connected network as classification head. The model was initialized with the pre-trained ImageNet weights and it was trained for 25 transfer-learning and 10 fine-tuning epochs. Note that the objective in this practical example is not necessarily to reach state-of-the-art performance of the AI component in the task at hand, but rather to assess the certification requirements of an AI model likely to be deployed for production.

A. Transparency

As first objective in the certification scheme, all stakeholders and the form of explanation each of them need about the AI system have to be identified (T1). This dictates the selection of suitable methods for assessing compliance. This application may require the model's predictions and decisions to be explainable only to domain experts, i.e. physicians. As these decisions directly affect patients, physicians cannot rely on predictions of black-box models without proper insight into the inference process (T2).

From this objective, the certification scheme identifies the technical methods necessary to assess this requirement (cf. Table I).

Since a DL model is used, which is not intrinsically explainable, post-hoc methods are required. Here, feature attribution maps can be used to produce explanations. These methods associate input features to how they perturb the prediction. LIME [32] and SHAP [29] are well-established methods for this purpose.

The effect of each pixel in the image on the output can generally be explored by perturbing the pixels as above, or by directly computing gradients of the prediction relative to the pixels, e.g. with [28], [30]. In all cases the result is an image; often called *saliency*, *attribution*, or *gradient* map (commonly superimposed over the original input image). This map indicates the regions of the image that supported the model's prediction, specifically for this use case, what visual characteristics led to the classification of the skin lesion as malignant or benign (T3).

In Figure 2, we show results of the SHAP (SHapley Additive exPlanations) method applied on a sample of a malignant melanoma. The explanation consists of a map with *Shapely values* for each pixel: a number centered around 0, negative values detrimental to, and positive values supportive of a specified target prediction. As such, SHAP is relatively intuitive to interpret for any lay person. Concretely, SHAP thus assists physician to discern what regions and properties such as shape or coloration of the imaged skin lesion most impacted the AI component's decision, be it in favor or against malignancy.

Although a physician may be most interested in local explanations during the operation of the AI component, another requirement for transparency is a thorough, statistical evaluation of the model behavior and documentation thereof. The certification scheme offers multiple options for technical methods to comply with this requirement.

If the model grants access to its internal structure, it is possible to use class activation maximization (CAM) [50] to retrieve a global explanation of the model behavior in form of an activation map. CAM is based on the gradient

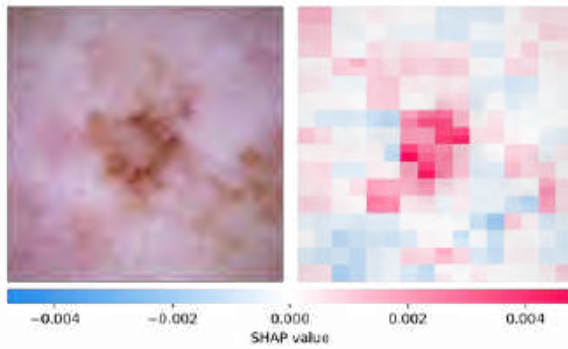


Fig. 2: The left panel shows a sample of a malignant melanoma and the corresponding Shapely value patch map on the right. A malignant lesion typically has a more asymmetric shape, a less defined border, and an irregular color profile. The Shapely values seem to support these heuristics overall.

ascent technique which is used in CNN feature visualization to maximize the response of a particular feature map. This is useful for the developer or assessor of the AI component to better understand the model's internal decision-making processes. In Figure 3, such activation maps are shown for two model variants, one properly trained (a), the other exhibiting unsatisfactory behavior (b). This highlights the practicability of this method for debugging and model assessment during the training or optimization phase of CNNs.

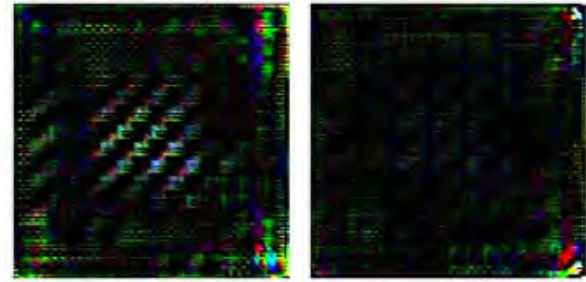
Alternatively, it could be sufficient to verify the expected behavior of a model on a small subsample of the test set which is representative of the whole data statistics. In such a context, the subsample instances are called *prototypes* and can in principle be extracted using various clustering algorithms, although ideally, prototypes are accompanied with *criticisms*, measuring how well a certain instance is represented by the prototypes; algorithms like ProtoDash [25] are more suited in this case. In combination with methods such as SHAP, the global behavior of a model can then be estimated through their evaluation on all prototypes and criticisms.

B. Reliability

In this example, there are numerous perturbations that pertain to the ODD, which should be covered according to the objectives defined in Table I (R2). We report two of them: one arising from the camera hardware (Gaussian Noise, GN) and another caused by environmental factors (Brightness Change, BC). Table II lists the parameter intervals for each perturbation that can be expected in real-world scenarios. These intervals were used for perturbation simulations on the ISIC2019 test set and all subsequent tests.

To illustrate the effect of the perturbations and their combination, Figure 4 shows samples with increasing noise or brightness from left to right.

For each type of perturbation, we applied various perturbation levels represented by different parameter values (e.g., with a step size of 0.025 for GN) in order to cover a wide range of potential scenarios within the ODD. Regarding BC,



(a) Accurate model

(b) Bad model

Fig. 3: Panel (a) shows the CAM map for a malignant classification of a properly trained model, and (b) of an improperly configured model. These maps explain the internal features of the model with arbitrary input. For this specific use case, it is evident that certain color channels in the central region of the image have the most impact in a malignant classification.

TABLE II: Definition of perturbations relevant to the ODD with specific parameter intervals for each perturbation type, taking into account combined perturbations.

Description	Perturbation type	Parameter interval
White noise on camera	GN $\mathcal{N}(\mu = 0.0, \sigma)$	$\sigma \in [0; 0.1]$
Different illumination	BC	$b \in [-0.3; 0.3]$
Perturbations combined	Both combined	$\sigma \wedge b$

we randomized the BCs for each pixel within a five percent deviation from the chosen level to create a more realistic perturbation. In the CI tests, we conducted 1000 simulations for each of these perturbation levels on every sample in the test set, employing the specified noise and/or brightness parameter value. Based on the simulated perturbation statistics, the mean accuracy and the 2-sided 95% CI were computed. Table III exemplifies the results for two different perturbation levels for each perturbation type.

This CI serves as an indicator of the robustness of the AI system. According to the objectives (R1), specific appropriate performance metrics (and their goals) are to be defined. Thus, the performance of the model can be evaluated by comparing the performance goals (e.g., for accuracy in this example) with the test results in Table III.

These tests also provide the basis for evaluating the uncertainty of the AI system. They allow both a graphical evaluation using the reliability diagram and a metric-based evaluation. The reliability diagram is derived based on the unperturbed test set, while the applied metrics are also evaluated based on the different perturbation levels.

The reliability diagram (a) in Fig. 5 illustrates the model calibration accuracy. The proximity of the results to the green dotted line indicates good calibration, while being below it reflects overconfidence compared to actual accuracy. In addition, the confidence diagram (b) gives further information about the distribution of samples for different confidence levels within the test set. Thereby, it becomes evident that the evaluated

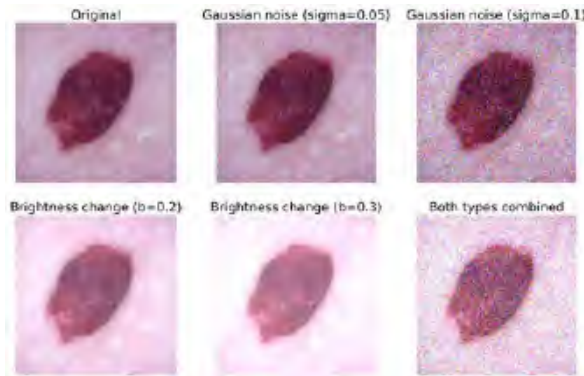


Fig. 4: For skin lesion detection, a dermatology camera is typically used, where perturbations such as the GN (caused by electronic circuits) or BC (caused by environmental factors) occur. All relevant types of perturbations can arise at the same time.

TABLE III: Results of the CI tests (simulated 1000 times for each sample of the test set) summarized by their mean accuracy and respective 2-sided 95% CI, along with the ECE and multi-class Brier score for uncertainty evaluation.

Perturbation type	Mean accuracy (95% CI)	ECE	Brier score
GN ($\sigma = 0.05$)*	0.713 ± 0.13	0.10	0.29
GN ($\sigma = 0.1$)**	0.697 ± 0.21	0.16	0.36
BC ($b = 0.1$)*	0.726 ± 0.07	0.09	0.31
BC ($b = 0.2$)**	0.711 ± 0.15	0.14	0.45
Both combined*	0.691 ± 0.29	0.18	0.48
Both combined**	0.675 ± 0.35	0.21	0.57

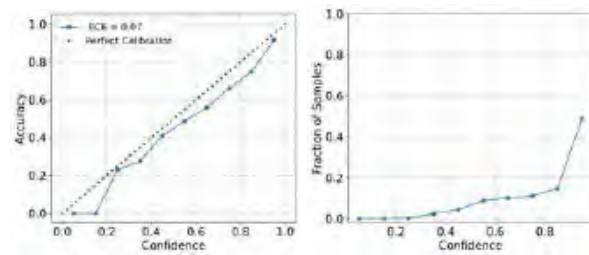
predictions of the AI system consist mostly of predictions with a confidence level above 0.7.

Beyond the graphical evaluation, the uncertainty metrics ECE and multi-class Brier score, introduced in Section V are evaluated on the perturbed test set data for the different perturbation levels used within the CI tests (cf. Table III), as required in the last objective of the certification scheme (R3). These results indicate that the CI system under test has increasing model uncertainty (increasing ECE and Brier scores) when processing perturbed data.

VII. CONCLUSIONS

Certifying the trustworthiness of AI systems is crucial to ensure their safety and for enabling compliance with regulatory requirements. AI-related ethical guidelines and frameworks for regulation provide high-level objectives for various dimensions such as transparency, reliability, safety and others. However, they often lack detailed information on the metrics, criteria, processes, and methods that should be used to assess compliance.

We present here our ongoing work on the development of a certification scheme, specifically tailored to the proposed requirements of the imminent EU AI regulation and based on international standards and guidance documents. The certification scheme extends previous work by explicitly bridging the



(a) Reliability diagram

(b) Confidence diagram

Fig. 5: The reliability diagram (a) visualizes the model calibration capability, while the confidence diagram (b) depicts the fraction of samples for the confidence levels.

high-level regulatory requirements to state-of-the-art technical methods for reliably assessing AI trustworthiness.

This paper demonstrates the certification scheme using an example case for a small set of objectives for transparency and reliability. The entire certification scheme thoroughly covers the full set of objectives and trustworthiness dimensions. In the next stages of our research, we will extend the scheme to four dimensions of trustworthiness and validate it on more real-world use cases in the limited and high-risk categories. Yet, several challenges are still ahead: as research in the topic continues, emergent technical methods need to be tested for applicability in real-world scenarios. We advise researchers the use of datasets reflecting real-world conditions and integration of trustworthiness objectives in the validation of these methods. Moreover, the scarcity of appropriate metrics in these cases is concerning and has to be addressed.

As organizations will be required to comply with the coming AI-related regulations, the certification scheme can support developers, compliance officers, and certification bodies in appropriately evaluating AI-based systems, taking informed decisions, and enabling certification of innovative AI-based products.

ACKNOWLEDGMENT

This work was co-financed by Innosuisse (101.650 IP-ICT).

REFERENCES

- [1] A. Jobin, M. Ienca, and E. Vayena, "The Global Landscape of AI Ethics Guidelines," *Nature Machine Intelligence*, vol. 1, no. 9, pp. 389–399, 2019. [Online]. Available: <https://doi.org/10.1038/s42256-019-0088-2>
- [2] Council of European Union. (2021) Council Regulation (EU) No Com/2021/206 Final. [Online]. Available: <https://eur-lex.europa.eu/leg al-content/EN/TXT/?uri=CELEX:52021PC0206>
- [3] —, (2023) Artificial Intelligence Act: Council and Parliament Strike a Deal on the First Rules for AI in the World. [Online]. Available: <https://www.consilium.europa.eu/en/press/press-releases/2023/12/09/art ificial-intelligence-act-council-and-parliament-strike-a-deal-on-the-fir st-worldwide-rules-for-ai/>
- [4] The White House. (2023) Fact Sheet: President Biden Issues Executive Order on Safe, Secure, and Trustworthy Artificial Intelligence. [Online]. Available: <https://www.whitehouse.gov/briefing-room/statements-relea ses/2023/10/30/fact-sheet-president-biden-issues-executive-order-on-s afe-secure-and-trustworthy-artificial-intelligence/>
- [5] ISO. (2023) ISO/IEC JTC 1/SC 42 Artificial Intelligence. [Online]. Available: <https://www.iso.org/committee/6794475.html>

- [6] IEEE Standards Association. (2023) IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems. [Online]. Available: <https://standards.ieee.org/industry-connections/ec/autonomous-systems/>
- [7] NIST. (2023) NIST Technical AI Standards. [Online]. Available: <https://www.nist.gov/artificial-intelligence/technical-ai-standards>
- [8] DIN, DKE. (2023) Artificial Intelligence Standardization Roadmap. [Online]. Available: <https://www.dke.de/en/areas-of-work/core-safety/standardization-roadmap-ai>
- [9] M. Poretschkin *et al.*, “Leitfaden Zur Gestaltung Vertrauenswürdiger Künstlicher Intelligenz,” Fraunhofer IAIS, Tech. Rep., 2021. [Online]. Available: <https://www.iais.fraunhofer.de/de/forschung/kuenstliche-intelligenz/ki-pruefkatalog.html>
- [10] LNE. (2023) Certification of Processes for AI. [Online]. Available: <https://www.lne.fr/en/service/certification/certification-processes-ai>
- [11] IEEE. (2022) IEEE CertifAIED. [Online]. Available: <https://engagestandards.ieee.org/ieeecertifai.html>
- [12] G. Soudain, “First Usable Guidance for Level 1 Machine Learning Applications: A Deliverable of the EASA AI Roadmap,” 2021. [Online]. Available: <https://www.easa.europa.eu/en/downloads/134357/en>
- [13] Trusted-AI LF AI Foundation. AI Fairness 360 (AIF360). [Online]. Available: <https://github.com/Trusted-AI/AIF360>
- [14] —. Adversarial Robustness Toolbox (ART). [Online]. Available: <https://github.com/Trusted-AI/adversarial-robustness-toolbox>
- [15] —. AI Uncertainty Quantification 360 (UQ360). [Online]. Available: <https://github.com/Trusted-AI/UQ360>
- [16] —. AI Explainability 360 (AIX360). [Online]. Available: <https://github.com/Trusted-AI/AIX360>
- [17] N. Kokhlikyan, V. Miglani, M. Martin *et al.*, “Captum: A Unified and Generic Model Interpretability Library for Pytorch.”
- [18] Seldon. Alibi Explain. [Online]. Available: <https://github.com/Seldon/Oalibi>
- [19] Microsoft. InterpretML. [Online]. Available: <https://github.com/interpretml/interpret>
- [20] PAIR-code. What-If Tool. [Online]. Available: <https://github.com/pair-code/what-if-tool>
- [21] W. D. Nothwang, M. J. McCourt, R. M. Robinson *et al.*, “The Human Should Be Part of the Control Loop?” in *2016 Resilience Week (RWS)*. IEEE, 2016. [Online]. Available: <https://doi.org/10.1109/RWEEK.2016.7573336>
- [22] C. Molnar, G. Casalicchio, and B. Bischl, “Interpretable Machine Learning – a Brief History, State-Of-The-Art and Challenges,” *Communications in Computer and Information Science*, pp. 417–431, 2020. [Online]. Available: https://dx.doi.org/10.1007/978-3-030-65965-3_28
- [23] W. Zhao, S. Alwidian, and Q. H. Mahmoud, “Adversarial Training Methods for Deep Learning: A Systematic Review,” *Algorithms*, vol. 15, no. 8, p. 283, 2022. [Online]. Available: <https://doi.org/10.3390/a15080283>
- [24] V. Arya, R. K. E. Bellamy, P.-Y. Chen *et al.*, “One Explanation Does Not Fit All: A Toolkit and Taxonomy of AI Explainability Techniques,” 2019. [Online]. Available: <https://doi.org/10.48550/arXiv.1909.03012>
- [25] K. S. Gurumoorthy, A. Dhurandhar, G. Cecchi, and C. Aggarwal, “Efficient Data Representation by Selecting Prototypes with Importance Weights,” 2017. [Online]. Available: <https://doi.org/10.48550/arXiv.1707.01212>
- [26] A. Kumar, P. Sattigeri, and A. Balakrishnan, “Variational Inference of Disentangled Latent Concepts from Unlabeled Observations,” 2017. [Online]. Available: <https://doi.org/10.48550/arXiv.1711.00848>
- [27] B. M. Greenwell, B. C. Boehmke, and A. J. McCarthy, “A Simple and Effective Model-Based Variable Importance Measure,” 2018. [Online]. Available: <https://doi.org/10.48550/arXiv.1805.04755>
- [28] K. Simonyan, A. Vedaldi, and A. Zisserman, “Deep inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps,” 2013. [Online]. Available: <https://doi.org/10.48550/arXiv.1312.6034>
- [29] S. M. Lundberg and S.-I. Lee, “A Unified Approach to Interpreting Model Predictions,” in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio *et al.*, Eds. Curran Associates, Inc., 2017, pp. 4765–4774. [Online]. Available: <http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf>
- [30] A. Ali, T. Schnake, O. Eberle *et al.*, “XAI for Transformers: Better Explanations through Conservative Propagation,” 2022. [Online]. Available: <https://doi.org/10.48550/arXiv.2202.07304>
- [31] D. K. I. Weidele, H. Strobelt, and M. Martino, “Deepling: A Visual Interpretability System for Convolutional Neural Networks,” 2019. [Online]. Available: https://mlsys.org/Conferences/2019/doc/2019/demo_22.pdf
- [32] M. Ribeiro, S. Singh, and C. Guestrin, “‘‘Why Should I Trust You?’’: Explaining the Predictions of Any Classifier,” in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*. Association for Computational Linguistics, 2016. [Online]. Available: <https://arxiv.org/abs/1602.04938>
- [33] A.-p. Nguyen and M. R. Martínez, “On Quantitative Aspects of Model Interpretability,” 2020. [Online]. Available: <https://doi.org/10.48550/arXiv.2007.07584>
- [34] F. Bodria, F. Giannotti, R. Guidotti *et al.*, “Benchmarking and Survey of Explanation Methods for Black Box Models,” *Data Min. Knowl. Discov.*, vol. 37, no. 5, pp. 1719–1778, Jun. 2023. [Online]. Available: <https://doi.org/10.1007/s10618-023-00933-9>
- [35] N. Akhtar, “A Survey of Explainable AI in Deep Visual Modeling: Methods and Metrics,” 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2301.13445>
- [36] SAE. (2021) Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles. SAE J3016. [Online]. Available: https://www.sae.org/standards/content/j3016_202104/
- [37] D. Hendrycks and T. Dietterich, “Benchmarking neural network robustness to common corruptions and perturbations.”
- [38] D. R. Cox and D. V. Hinkley, *Theoretical statistics*. CRC Press, 1979.
- [39] T. Gehr, M. Mirman, D. Drachler-Cohen *et al.*, “AI2: Safety and Robustness Certification of Neural Networks with Abstract Interpretation,” in *2018 IEEE Symposium on Security and Privacy (SP)*, 2018, pp. 3–18. [Online]. Available: <https://doi.org/10.1109/sp.2018.00058>
- [40] G. Singh, T. Gehr, M. Püschel, and M. Vechev, “An Abstract Domain for Certifying Neural Networks,” *Proceedings of the ACM on Programming Languages*, vol. 3, no. POPL, pp. 1–30, 2019.
- [41] M. H. DeGroot and S. E. Fienberg, “The Comparison and Evaluation of Forecasters,” *Journal of the Royal Statistical Society: Series D (The Statistician)*, vol. 32, no. 1/2, p. 12, 1983.
- [42] A. Niculescu-Mizil and R. Caruana, “Predicting Good Probabilities with Supervised Learning,” in *Proceedings of the 22nd international conference on Machine learning*, ser. ICML ’05. ACM Press, 2005, pp. 625–632.
- [43] M. Pakdaman Naeini, G. Cooper, and M. Hauskrecht, “Obtaining Well Calibrated Probabilities Using Bayesian Binning,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, 2015.
- [44] G. W. Brier, “Verification of Forecasts expressed in Terms of Probability,” *Monthly Weather Review*, vol. 78, no. 1, pp. 1–3, 1950.
- [45] Y. Gal and Z. Ghahramani, “Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning,” in *Proceedings of The 33rd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, M. F. Balcan and K. Q. Weinberger, Eds., vol. 48. New York, New York, USA: PMLR, 20–22 Jun 2016, pp. 1050–1059. [Online]. Available: <https://proceedings.mlr.press/v48/gal16.html>
- [46] S. Depeweg, J.-M. Hernandez-Lobato, F. Doshi-Velez, and S. Udluft, “Decomposition of Uncertainty in Bayesian Deep Learning for Efficient and Risk-sensitive Learning,” in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. PMLR, 10–15 Jul 2018, pp. 1184–1193. [Online]. Available: <https://proceedings.mlr.press/v80/depeweg18a.html>
- [47] C. Huyen, *Designing Machine Learning Systems*. USA: O’Reilly Media, 2022.
- [48] P. Tschandl, N. Codella, B. N. Akay *et al.*, “Comparison of the Accuracy of Human Readers Versus Machine-Learning Algorithms for Pigmented Skin Lesion Classification: An Open, Web-Based, International, Diagnostic Study,” vol. 20, no. 7, pp. 938–947.
- [49] M. Tan and Q. Le, “EfficientnetV2: Smaller Models and Faster Training,” in *International conference on machine learning*. PMLR, 2021, pp. 10 096–10 106. [Online]. Available: <https://proceedings.mlr.press/v139/tan21a.html>
- [50] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, “Learning Deep Features for Discriminative Localization,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2921–2929.

Advances in Reliability,
Safety and Security

ESREL 2024
Monograph Book Series

A Certification Scheme for Artificial Intelligence based Systems

Yann Billeter^a, Stefan Brunner^a, Ricardo Chavarriaga^a, Philipp Denzel^a, Oliver Forster^a, Carmen Mei-Ling Frischknecht-Gruber^a, Monika Reif^a, Frank-Peter Schilling^a, Joanna Weng^a
Arman Iranfar^b, Marco Repetto^b

^aZurich University of Applied Sciences ZHAW, Winterthur, Switzerland

^bCertX AG, Fribourg, Switzerland

Organiser use only: Received date; revised date; accepted date

Abstract

Artificial Intelligence (AI) and machine learning (ML) algorithms are making an impact in an increasing number of industries. AI models differ from conventional software due to their probabilistic decision-making process, with a heavy reliance on training data quantity and quality. Ensuring the trustworthiness of AI-based systems (AIS), including dimensions such as reliability and transparency, is becoming increasingly vital due to their widespread adoption. As regulatory standards are put in place, it becomes essential to have practical guidelines for certification. In this paper, we present an ongoing effort to develop a validated Certification Scheme for AIS. This scheme encompasses distinct objectives, criteria, and corresponding measures, as well as specific metrics and technical methods which support the implementation of trustworthy AI. A critical aspect of this scheme is the establishment of a clear connection between the set of requirements and the validated ML algorithms and methods used to evaluate the compliance of AIS. We provide a tangible example of the workflow for the reliability dimension on a hypothetical real-life use case: employing the Yolo5 model for the detection of construction vehicles in a diverse image dataset of construction sites. This example demonstrates the step-by-step process of the Certification Scheme from establishing initial requirements to selecting and applying technical methods for two example objectives.

Keywords: Artificial intelligence, machine learning, certification, trustworthiness, object detection

1. Introduction

Legislators and authorities are working to establish a high level of trust in Artificial Intelligence (AI). On Friday, December 8, 2023 – after months of intensive negotiations – the European Parliament and Council reached political agreement on the European Union's Artificial Intelligence Act "EU AI Act" (European Commission, 2021). Nevertheless, even with these forthcoming regulations in the next 2-3 years, there remains a notable absence of practical guidelines and translation into specific methods and practices for effectively evaluating the trustworthiness of an AI-based systems (AIS). The deployment of AI technologies that are not fully understood and reliable can cause severe harm to society, for example by excluding minorities due to biases, causing direct physical injuries, or through misdiagnosis of patients in the health sector. The goal of the ongoing project CertAInty, which is carried out by a consortium formed by Zurich University of Applied Sciences ZHAW (Winterthur, CH) and CertX AG (Fribourg, CH), is to develop a Certification Scheme as practical guide for both developers and regulatory bodies to evaluate and certify the trustworthiness of AIS.

A Certification Scheme is a framework for the certification of AIS, including specific objectives, criteria and corresponding means of compliance. It covers the entire life cycle of an AIS encompassing data acquisition, model development and testing, deployment, and operation. Furthermore, it is designed to be inclusive, considering the perspectives of all stakeholders such as developers, end-users, auditors, and regulatory authorities.

The novelty of this certification scheme is to bridge the gap between regulations and technical standards on the one hand and concrete scientific and technical methods to verify properties of machine learning models on the other. It will cover the following certification dimensions: reliability, transparency, autonomy and control and safety. In this paper, initial results for the reliability dimension are presented.

2. Background

Numerous national and international organizations are actively engaged in initiatives aimed at fostering trust in AI. The LNE has established an AI certification program that sets impartial and objective criteria for trustworthy AI systems. This program covers essential aspects like ethics, safety, transparency, and privacy (LNE, 2023). Several organisations and initiatives such as ISO/IEC (ISO, 2023) and NIST (NIST, 2023) are currently working on developing relevant AI standards. In addition, IEEE is in the process of creating a certification program focused on evaluating transparency, accountability, bias, and privacy in AIS (IEEE, 2022).

Additionally, EASA has released a detailed guideline for the safe application of machine learning in aviation (EASA, 2023). This guide assists aviation industry players in the development and implementation of ML systems, especially those with low levels of automation. It covers the full lifecycle, from development to operational use. Moreover, DIN/DKE offers comprehensive recommendations for standardizing AI, aiming to establish a common language and principles for development, use, and certification (DIN, DKE, 2023).

The Fraunhofer Institute has contributed to increasing trust in AI by developing guidelines for designing trustworthy AIS (Poretschkin et al., 2021). Their AI catalogue assesses AIS trustworthiness across six dimensions, including fairness, autonomy and control, transparency, reliability, safety/security, and privacy.

Method toolboxes and evaluation frameworks are vital for ensuring transparent, explainable, and robust AI systems and a variety is already available. Companies like IBM and Seldon have developed toolboxes, such as AIX360 (Bellamy et al., 2018) and Alibi (Klaise et al., 2021; Van Looveren et al., 2019), featuring methods for explainability and uncertainty quantification. The Adversarial Robustness Toolbox (ART) is another framework focused on evaluating the adversarial robustness of neural networks, incorporating different types of attacks (Nicolae et al., 2018).

3. Certification Scheme

The proposed Certification Scheme for AIS will cover the following four trustworthiness dimensions: **(1) Reliability:** Assesses the AIS ability to perform consistently under varying conditions, and be robust against errors, biases, or potential security threats; **(2) Transparency:** Essential for allowing different stakeholders to comprehend the AI system's decision-making processes, thereby facilitating informed trust and reliance on the system; **(3) Safety:** Particularly crucial in safety critical domains like healthcare or autonomous vehicles, to prevent any adverse outcomes or unintended consequences of AIS's operations; **(4) Autonomy and Control:** Involves understanding the level of the AIS's independence, ranging from systems requiring human oversight ('Human-in/on-the-Loop') to those operating autonomously.

The Certification Scheme uses a risk-based approach and consists of two parts: (a) the framework itself that summarizes objectives, criteria, and various means of compliance (MOC) needed to assess the trustworthiness of an AIS; and (b) guidance linking these requirements to a set of technical and scientific methods for assessing relevant characteristics of AIS. It provides a complete workflow to identify and apply methods and processes for assessing compliance with the emerging AI regulations.

To define the Certification Scheme, an iterative approach is employed. Initially, a draft Certification Scheme is created, outlining the main objectives for achieving conformity with EU legislation. Then, the associated means of compliance are defined, distinguishing between metrics, processes, documentation, and methods. The next step is to identify, test and refine these technical and algorithmic methods in order to achieve sufficient compliance with the certification objectives. The choice of appropriate methods can vary based on several factors, including the data type, model type, the life cycle phase, and the stakeholders involved. The iterative approach ensures the certification scheme is agile, reliable, and effective. In the first version of the Certification Scheme, we included two dimensions, namely Transparency (encompassing 29 objectives and 100 MOC) and Reliability (44 objectives and 156 MOC).

In the following, we focus on the reliability dimension and show the path from the objectives to the technical methods for two example objectives and MOCs.

Objective 1:

O1: The applicant should define performance metrics to evaluate the AIS performance and reliability for the regular case.

MOC: The applicant should define a suitable set of performance metrics for each high-level task to evaluate the AIS performance and reliability.

MOC: The applicant should define the expected performance with training, validation, and test data sets.

MOC: The applicant should provide a comprehensive justification for the selection of metrics.

Objective 2:

O2: The coverage of the application boundary must be formalized and quantified, if possible, and application-specific target ranges for the coverage of the application boundary must be defined.

MOC: The applicant should formalize and quantify the coverage of the Operational Design Domain (ODD, see below), especially when data point perturbations are influenced or described by factors not fully addressed for the ODD.

MOC: The application boundary may be graded differently depending on the nature and severity of the perturbation.

4. Reliability assessment of AI systems

Reliability in AI systems, as discussed in this paper, is the capacity of an AIS to consistently execute its intended functions. Robustness, on the other hand, is about maintaining performance and functionality in the face of disturbances. The requirements discussed in the following refer to different AI development lifecycle phases such as requirements definition, data acquisition, model training and verification, inference model verification and integrated AIS verification.

Operational Design Domain (ODD): While the term ODD is deeply rooted in autonomous vehicle technology (SAE International, 2021) its utility may be extended to a broader range of AIS applications. Essentially, an ODD defines the specific conditions under which a given AIS should operate safely and effectively. This concept is essential to ensure that the AIS operates within its designed capabilities and environmental constraints.

To clarify this concept in the context of autonomous driving, the ODD includes several operational parameters. These may include the types of roads the vehicle can navigate, the geographic areas in which it can operate, the static and dynamic objects the system may encounter, the weather conditions it can handle, and even the times of day or night during which it will function. For example, an autonomous vehicle's ODD may specify that it can only operate on clear days, on highways, and not on one-way streets.

The level of detail within an ODD varies depending on the intended audience. For developers and engineers, an ODD may need to include highly specific technical details. Thus, the ODD is the basis for defining the input space for an AIS, which can then be categorized into regular, robustness, and out-of-domain (OOD) cases. In the regular case, the system should be able to reliably handle small disturbances, while in the robustness case it should be able to robustly cope with large disturbances. However, in OOD cases, where data falls outside the application domain, the system may not perform adequately, leading to potential errors.

4.1. Reliability key areas

In the reliability dimension of the certification scheme for AIS, which is firmly grounded in the definition of the ODD, we consequently distinguish between 4 different key areas, as detailed in Table 1.

In the first area, the **regular case**, which focuses on the standard application domain, several essential actions are required to ensure reliable system performance. A thorough data coverage assessment is vital to verify that the data set accurately represents the application domain. Data augmentation is often necessary to increase the diversity of the dataset and prepare the system for a wider range of situations during training or testing. The adaptability and accuracy of the system must be evaluated using performance metrics with acceptable target values. At the same time, the application of loss metrics during the training phase is essential to improve model accuracy and minimize error rates. Conditioning the system to effectively handle common and environmental perturbations is another critical step to improve operational resilience. Finally, it's important to develop strategies that ensure reliable generalization on unseen data within this standard application domain and to implement mitigation measures for misjudgement.

The **robustness case** extends these principles by maintaining data coverage evaluation and augmentation while emphasizing robustness, especially under challenging conditions such as edge and corner cases. Performance metrics are evaluated under these stringent conditions, and a systematic vulnerability assessment is performed to identify and address potential weaknesses. The robustness case also addresses common perturbations and extreme environmental scenarios, incorporating strategies to mitigate adversarial attacks and exploring generalization beyond typical operations.

The **OOD case** further expands the scope to include scenarios outside of the regular and robust cases. Data augmentation is adapted for OOD data, and special attention is given to catching errors at the input. Vulnerability assessment remains vital, along with protection against adversarial attacks and exploration of generalization outside the expected operational domain.

Finally, the **Uncertainty Estimation** aspect involves a detailed uncertainty assessment, where appropriate uncertainty metrics need to be defined and applied. This stage focuses on quantifying both intrinsic and extrinsic uncertainties and developing mitigation measures to address uncertainties in decision processes.

Table 1. Reliability key areas with their main actions

Regular Case	Robustness Case	OOD Case	Uncertainty Estimation
Data coverage assessment	Data coverage assessment	Data coverage assessment	Uncertainty assessment
Data augmentation	Data augmentation	Data augmentation	Uncertainty metrics
Performance metrics	Performance metrics	Catching errors at input	Uncertainty estimation
Loss metrics	Vulnerability assessment	Vulnerability assessment	
Common perturbations	Common perturbations	Common perturbations	
Environmental perturbations	Environmental perturbations	Environmental perturbations	
	Adversarial attacks	Adversarial attacks	
Generalization within regular case	Generalization and exploration within robustness case	Generalization and exploration outside expected operation	
Mitigation measures for misjudgement	Mitigation measures for misjudgement	Mitigation measures for detected errors	Mitigation measures for uncertainties

In addition, process steps such as evaluating the model architecture, implementing optimization techniques such as pruning or quantization which are often required in order to deploy the AI model on an edge device, and ensuring reproducibility are important, as are conducting regular assessments and meticulously documenting all activities.

4.2. Metrics and technical methods for reliability assessment

We identified more than 55 metrics and 95 methods relevant for the certification of the reliability dimension based on a comprehensive review of the state of the art. After further analysis, we selected a subset of 35 metrics and 50 methods for empirical tests using as criteria: (i) their relevance to the intended applications; (ii) their applicability for certification regarding execution time, reliance on available information and computational costs.

The selection of the appropriate **evaluation metrics** is essential for an AIS and varies significantly across application domains and the objectives of the model. For example, in regression tasks, metrics such as (Mean) Squared Error (MSE) and (Mean) Absolute Error (MAE) are often used to evaluate prediction accuracy. In classification scenarios, a variety of metrics such as (Mean) Accuracy, Precision, Recall, F1 Score, and Area Under the Curve (AUC) value are used to measure how effectively the model categorizes the data. Ranking algorithms typically use Mean Reciprocal Rank (MRR) and Discounted Cumulative Gain (DCG) to determine the relevance of ranked outputs, while clustering models use measures such as Silhouette Value and Adjusted Mutual Information (AMI) Score to evaluate data groupings. In specialized areas such as computer vision, metrics such as Mean Intersection over Union (mIoU) and Mean Average Precision (mAP) are used, while Natural Language Processing (NLP) often relies on Perplexity Score and Bilingual Evaluation Understudy (BLEU) Score for language models and translation quality.

To assess and improve the reliability of AIS, **technical methods** are provided that are tailored to specific challenges such as common perturbations, environmental factors, adversarial inputs, and uncertainty. For common perturbations, techniques such as injecting Gaussian noise into data sets or applying geometric transformations (e.g., rotations, scaling) test the stability of the system against routine data variations. Environmental perturbations are addressed through methods such as varying lighting conditions in image processing tasks or simulating different climatic conditions through scenario-based testing and real-world data augmentation, enriching the training dataset with diverse environmental examples to increase adaptability and resilience. To counter adversarial threats, approaches such as the Fast Gradient Sign Method (FGSM) or Generative Adversarial Networks (GANs) are used

to create challenging adversarial examples that enhance the system's ability to withstand malicious input. Uncertainty assessment incorporates Bayesian methods for probabilistic modelling and Monte Carlo simulations to evaluate the model's response to a wide range of inputs, thereby assessing predictive uncertainty. Coverage assessment is addressed through techniques such as formal verification, which uses logical and mathematical proofs to confirm that the system meets specified criteria, and model coverage analysis, which assesses whether all aspects of the system have been thoroughly tested against a comprehensive range of scenarios, including edge cases.

It is important to note that the above mentioned are examples, and a wider range of other metrics and methods may be more appropriate depending on the context and nuances of the specific problem. Therefore, a thorough evaluation is required to select the most appropriate metric for the certification, considering the objectives of the model, the nature of the data, and the desired outcomes.

5. Real-life use case: Vehicle detection on construction sites

In the following, we consider a hypothetical real-life use case in which a computer vision model in an AIS is required to detect specific kinds of vehicles and equipment on construction sites, as depicted in specific image datasets from the Roboflow 100 (RF100) benchmark¹. RF100 is a crowdsourced, open-source benchmark which contains over 90'000 image datasets spanning a wide range of domains. This initiative, sponsored by Intel, is designed to provide a comprehensive and diverse benchmark for machine learning object detection models, allowing them to be tested across a wide range of real-world scenarios and increase their generalizability. We selected three of them, suited for the training of model: the *Excavators*², *Heavy Equipment*³, and *Construct.AI.v2.BB*⁴ datasets. The compiled dataset contains 12963 images, featuring 7 classes of vehicles, and over 15'000 class instances (see Table 2 for the full list). The train/evaluation/test split is 85/10/5%, respectively, preserving the class distributions in each partition.

Table 2. The dataset's class instance distribution in numbers and percentages of the total. Note that an image may feature multiple instances.

Class	Instances	Percentage (%)
dump truck	3470	23
loader	2799	19
excavator	3793	25
roller	2843	19
mobile crane	456	3
bulldozer	918	6
grader	787	5

The YOLOv5 architecture was chosen for the object detection component of the AIS, as it presents a solid choice for a wide variety of object detection tasks with (nearly) state-of-the-art performance. It was designed by Ultralytics to be fast, accurate, and easy to use, with flexible open-source licensing. It is the fifth iteration of the original YOLO network (You Only Look Once; Redmon et al. 2015, 2016, 2018), which, as its name suggests, connects the procedure of predicting class labels (classification) with bounding boxes (detection) in an end-to-end differentiable network. Thus, it approaches the object detection task as a regression problem by spatially separating bounding boxes and associating scores to each of them. This allows the model to detect multiple instances in a single image, and in fact often predicts multiple box candidates for a single instance in which case a thresholding algorithm filters out the relevant predictions.

6. Technical assessment of the use case

The following section addresses objectives **O1 (Definition of performance metrics and evaluation of the AIS's performance and reliability)** and **O2 (Formalization and quantification of application coverage)** from the certification scheme in section 3, specifically in the context of Operational Design Domain (ODD) coverage. This is demonstrated based on the use case described in the previous section.

¹ <https://www.rf100.org/>

² <https://universe.roboflow.com/mohamed-sabek-6zmr6/excavators-cwlh0>

³ https://universe.roboflow.com/kfu-ye4kz/heavy_equipment-ifagm

⁴ <https://universe.roboflow.com/andrew-hannell/construct.ai.v2.bb>

6.1. Objective 1: Definition of metrics and performance of the model

During the operation of the AIS, the object detection model may be required to identify a specific class, such as excavators or loaders, with a given minimum performance score. For this task there are several suitable metrics to evaluate the model's performance.

Intersection over Union (IoU) is a metric used in object detection to quantify the accuracy of a predicted bounding box compared to the ground truth (actual) bounding box. IoU is calculated by dividing the area of overlap between the predicted and ground truth bounding boxes by the area of the union of these two boxes. This ratio ranges from 0 to 1, where 1 indicates perfect overlap and 0 means no overlap. IoU is widely used in evaluating the performance of object detection models like YOLO.

Another metric widely established for object detection models is the **average precision (AP)** which generally measures the area under the precision-recall curve. Following the COCO (T.Y. Lin, 2014) standard, average precisions are evaluated for each class separately for a 101-point interpolated curve with an IOU threshold between 0.5 and 0.95 (in steps of 0.05) and afterwards averaged over all classes, yielding the mean average precision (mAP@50-95); analogously for mAP@50 at an IOU threshold of 0.5 corresponding to the VOC standard⁵. Both metrics incorporate the trade-off between precision and recall and take false positives and false negatives into account which makes them especially suited for object detection applications.

Based on the discussion above, we used precision, recall, mAP@50, and mAP@50-95 as metrics for evaluating the model performance on the test dataset. As the main objective of this work is the assessment of an AIS rather than the development of a highly tuned, optimized model, the performance goal was arbitrarily set to >90% precision on class average. Consequently, the model training was run using mostly default hyperparameter settings for almost 100 epochs until the mean precision for the validation dataset converged above this threshold. The final test set performance per class label is listed in Table 3.

Table 3. Test set performance of the model evaluated using the chosen metrics.

Metric \ Class	dump truck	loader	excavator	roller	mobile crane	bulldozer	grader
mAP@50	0.53	0.80	0.75	0.96	0.38	0.76	0.78
mAP@50-95	0.77	0.93	0.93	0.93	0.62	0.96	0.97
Recall	0.62	0.85	0.85	0.95	0.59	0.97	0.96
Precision	0.85	0.95	0.94	0.85	0.75	0.89	0.87

6.2. Objective 2: ODD Assessment

The reliability assessment focused on a subset of methods to cover both "regular case" and "robustness case" scenarios based on the defined ODD and expected hardware effects, covering normal operating conditions as well as less common and more challenging, but still expected conditions. The simulated perturbations are described in the following and the range of parameters for the different perturbations is summarized in Table 4.

6.2.1 ODD Simulation

Common perturbations: In order to evaluate the robustness of the ML model, several common perturbations are applied to the images of the dataset, in accordance with the objective O2. These perturbations, which address real-world challenges such as camera noise, motion effects, or other visual distortions, are simulated by various linear transformations and noise, and are designed to cover the scenarios within the ODD that the algorithms may encounter. The following common perturbations were applied to the data set:

- **Homogeneous noising:** involves adding a consistent layer of noise across the entire image, affecting its clarity.
- **Gaussian noise,** a statistical noise having a probability density function equal to that of the normal distribution, is often added to images to mimic random variations in pixel values.
- **Brightening** is another perturbation where the image's luminance is uniformly increased, which can lead to loss of detail in brighter areas.

⁵ Note that in the literature, mAP and AP are often used interchangeably, since the context usually implies the appropriate interpretation.

Table 4. Parameters of common and environmental perturbations with their relation to the ODD

Perturbation type	Name	Parameter	ODD reference
Common	Homogenous noise (ISO, 2021)	$k = [-0.2, 0.2]$	Camera type
Common	Gaussian noise (Cattin, P., 2016)	$\mu = 0.0; \sigma = [0.0, 0.2]$	Camera type
Common/ Environmental	Brightening (ISO, 2021)	$b = [-0.2, 0.2]$	Illumination (day, artificial)
Common/ Environmental	Contrast (Peli, E. 1990)	$c = [0.8, 1.2]$	Illumination (day, artificial)
Common	Rotation (ISO, 2021)	$\text{angle}[^{\circ}] = [-10, 10]$	Camera type
Common	Blurring (ISO, 2021)	$\sigma = [0.8, 1.5]$	Camera type
Common	Motion blur (vertical, horizontal) (Navarro, F., 2011)	Kernel size = $[5, 20]$	Camera type
Common/ Environmental	Blooming (ISO, 2021)	Value threshold = $[200, 240]$	Illumination (day, artificial).
Environmental	Snow (Von Bernuth, A., 2019)	Light to heavy (5 levels)	Snowfall
Environmental	Fog (Von Bernuth, A., 2019)	Light to heavy (5 levels)	Non-precipitating smaller water droplets
Environmental	Sand (Si, Y., 2022)	Light to heavy (5 levels)	Larger airborne particles
Environmental	Dust (Si, Y., 2022)	Light to heavy (5 levels)	Smaller airborne particles
Environmental	Rain (Tremblay, M., 2021)	Light to heavy (5 levels)	Rainfall

- **Contrast** specifically targets the range and intensity of the light-to-dark spectrum in an image. By increasing or decreasing contrast, it can either exaggerate or diminish the distinctiveness of features.
- **Rotation** involves turning the image around a central point, testing the algorithm's ability to recognize objects from different angles.
- **Blurring** is a common perturbation that simulates out-of-focus images, reducing their sharpness and detail.
- **Motion blur** replicates the effect of movement during image capture, creating streaks or blurs in the direction of motion.
- **Blooming** occurs when intense light sources in an image cause bleeding of light, affecting the visibility of adjacent areas.

Each perturbation introduced presents its own challenges, and evaluating its impact is critical to assessing the reliability of the ML model. This assessment is particularly important for determining the model's ability to effectively handle a range of real-world perturbations, as required by the objective O2. In Fig. 1, a selection of implemented common perturbations is shown.



Fig. 1. Examples of common perturbations

Environmental perturbations: In order to cover the environmental aspects within the objective O2, ODD simulation with a focus on environmental perturbations is necessary to evaluate the reliability of object detection algorithms under all relevant conditions as listed in Table 4. The following environmental perturbations were applied to the data set:

- **Fog**, creating a uniform, diffuse layer of moisture that can significantly reduce visibility. It tends to scatter light in all directions, resulting in a general loss of contrast in visual scenes and a white or grayish appearance.
- **Snow**: Assessing the impact of snow is vital as it can obscure road markings, signs, and even the roadway itself. Snow can lead to reduced visibility and can also interfere with the sensors and cameras.
- **Rain** can affect visibility and also interfere with sensors. Heavy rain can cause reflections, distortions, and other optical effects that challenge the perception systems of autonomous vehicles.
- **Sand**: Airborne sand can cause more localized and irregular visibility problems. Sand particles scatter light in a less uniform manner, resulting in heterogeneous visual effects. Visibility can be severely reduced, but the effect is more one of blocking and distorting light rather than uniformly scattering it.
- **Dust**: Finer dust particles can create a widespread haze that reduces overall visibility and image sharpness by uniformly scattering light. In addition, dust can give the air a brownish or yellowish hue and cause a heterogeneous effect on visibility, with some areas appearing clearer than others.

Fig. 2 shows a selection of implemented environmental perturbations. Different methods are used to augment the existing image dataset, including physics-based effects and a hybrid approach combining physics-based and learning-based methods as described in (Tremblay, M., 2021).

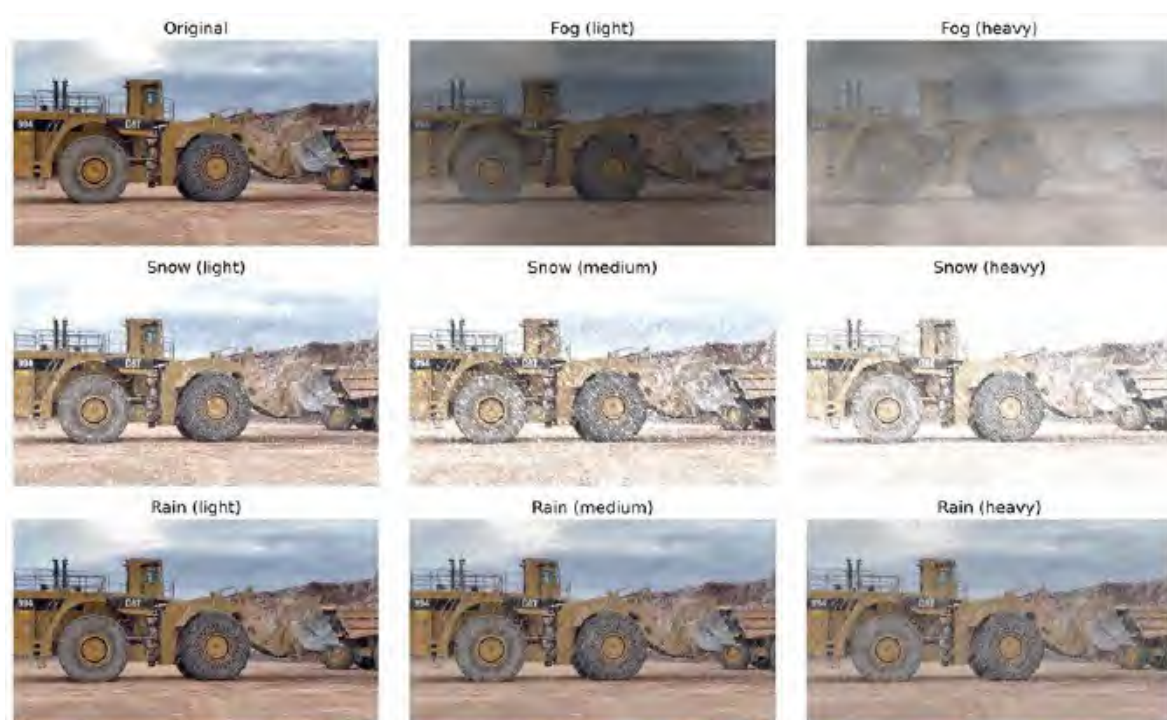


Fig. 2. Examples of environmental perturbations

Combined perturbations: The simulation of combined environmental and common perturbations offer a comprehensive evaluation of its robustness against multiple simultaneous challenges. Fig. 3 shows a selection of implemented combined perturbations. This approach provides a more accurate representation of the system's performance in diverse real-world scenarios.



Fig. 3. Examples of combined perturbations

6.2.2 ODD Assessment - Performance of the model

The AI model's performance was assessed using the previously defined metrics as function of the different simulated perturbations to measure how effectively the computer vision model can detect the specified vehicles and equipment under the formalized ODD conditions. Results show significant impact on the model performance, especially in the case of combined perturbations, and emphasize the need for rigorous testing using simulations to enhance the model's robustness in real-world applications. Based on the results, as a next step, specific target ranges for the chosen performance metrics need to be established. These targets set the expected performance levels for the model across diverse scenarios within the ODD, for instance different noise levels, and serve as benchmarks for future evaluations and enhancements. Fig. 4 summarizes the described certification workflow from the initial objectives to the robustness assessment. A complete evaluation of the application boundary would include a broader variety of technical methods, for example also adversarial attacks, assessments of concept and model drifts etc. not included in this analysis.

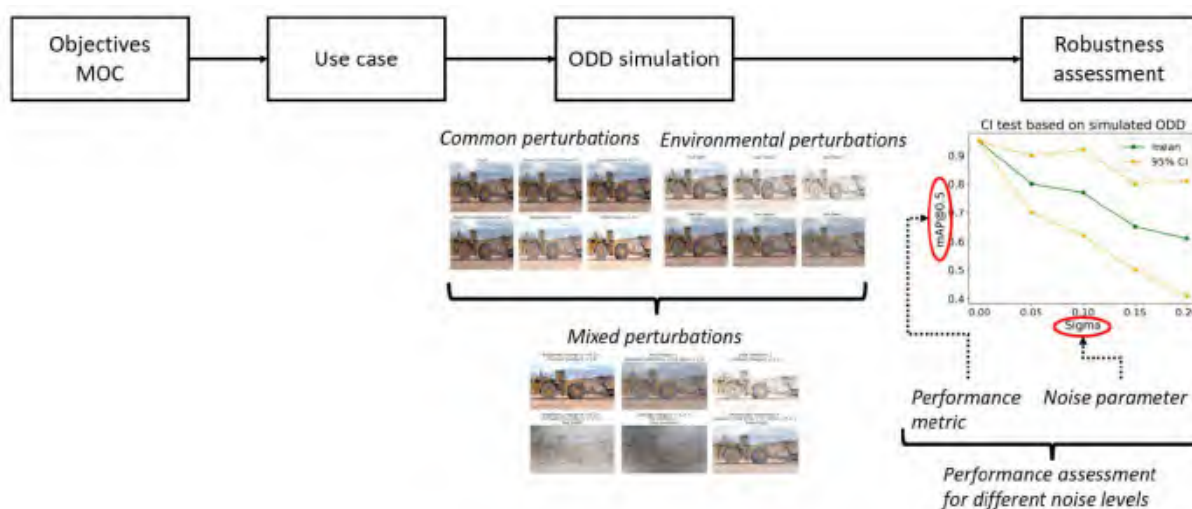


Fig. 4. Certification Workflow

7. Conclusion

Upcoming laws and regulations drive the need for established procedures to demonstrate and verify the trustworthiness of AI systems. In this paper we presented the development of an AI Certification Scheme which will cover the AI trustworthiness dimensions of reliability, transparency, autonomy and control, and safety. The scheme consists of two parts, namely the framework itself that summarizes objectives, criteria, and various means of compliance needed to assess the trustworthiness of an AIS, as well as guidance linking these requirements to a set of technical methods and metrics for assessing relevant characteristics of AIS. Objectives are defined based on the current legislation and the state of the art. Then, means of compliance to achieve these objectives are specified, distinguishing between criteria and metrics, processes, documentation, and methods to comply with the objectives. We demonstrated the application of this newly developed AIS Certification Scheme to a real-life use case for the reliability dimension, namely an AIS using a computer vision system to detect vehicles on construction sites. Starting with clear objectives, the workflow shows how technical methods and metrics are selected and applied to this use case. Performance metrics are defined and investigated as a function of various simulated perturbations, as defined in the Operational Design Domain (ODD). Besides the reliability dimension discussed in this paper, the final Certification Scheme will cover other dimensions such as transparency, autonomy and control and safety, and be applicable to a wide range of AIS, providing a guideline towards future certification of AI systems.

Acknowledgement: This work was co-financed by Innosuisse (101.650 IP-ICT).

References

- European Commission. 2021. Proposal for a Regulation of the European Parliament and of the Council laying down harmonised rules on artificial intelligence (Artificial Intelligence Act) and amending certain Union legislative acts. Official Document. Available at: [Link](#)
- LNE. 2023. Certification of Processes for AI. Journal of AI Process Certification 2023. Available at: [Link](#)
- IEEE Standards Association. 2023. IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems. Journal of Ethics in AI and Autonomous Systems 2023(1). Available at: [Link](#)
- NIST. 2023. NIST Technical AI Standards. Journal of Artificial Intelligence Standards 2023(1). Available at: [Link](#)
- DIN, DKE. 2023. Artificial Intelligence Standardization Roadmap. Journal of AI Standardization 2023. Available at: [Link](#).
- EASA, EASA Concept Paper: First usable guidance for Level 1 & 2 machine learning applications, February 2023, Issue 02
- Poretschkin, M., et al. 2021. Leitfaden zur Gestaltung vertrauenswürdiger Künstlicher Intelligenz (KI-Prufkatalog). Technical Report, Fraunhofer-Institut für Intelligente Analyse- und Informationssysteme IAIS. Available at: [Link](#)
- Bellamy, R.K.E., et al. 2018. AI Fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias. [arXiv:1810.01943](#).
- Klaise, J., A.V. Loooveren, G. Vacanti, and A. Coca. 2021. Alibi explain: Algorithms for explaining machine learning models. Journal of Machine Learning Research 22
- Nicolae, M.-I., et al. 2018. Adversarial robustness toolbox v1.0.0. [arXiv:1807.01069](#)
- SAE International. 2021. Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles. SAE J3016_202104. Available at: [Link](#)
- Redmon, J., et al. 2015. You Only Look Once: Unified, Real-Time Object Detection. [arXiv:1506.02640](#)
- Redmon, J., et al. 2016. YOLO9000: Better, Faster, Stronger. [arXiv:1612.08242](#)
- Redmon, J., et al. 2018. YOLOv3: An Incremental Improvement. [arXiv:1804.02767](#)
- Lin, T.Y., et al. 2014. Microsoft COCO: Common Objects in Context. ECCV 2014: 13th European Conference. [arXiv:1405.0312](#).
- International Organization for Standardization. 2021. ISO/IEC 24029-1:2021 - Artificial Intelligence - Robustness of Neural Networks Part 1: Overview. Available at: [Link](#)
- Cattin, P. 2016. Image Restoration: Introduction to Signal and Image Processing. MIAC, University of Basel.
- Peli, E. 1990. Contrast in complex images. Journal of the Optical Society of America A, 7(10), 2032-2040.
- Navarro, F., Serón, F. J., Gutierrez, D. 2011. Motion Blur Rendering: State of the Art. Computer Graphics Forum, Vol. 30, Issue 1.
- Von Bernuth, A., Volk, G., Bringmann, O. 2019. Simulating photo-realistic snow and fog on existing images for enhanced CNN training and evaluation. 2019 IEEE Intelligent Transportation Systems Conference (ITSC), 41-46.
- Tremblay, M., Halder, S. S., De Charette, R., Lalonde, J.-F. 2021. Rain rendering for evaluating and improving robustness to bad weather. International Journal of Computer Vision, 129, 341-360.
- Si, Y., Yang, F., Guo, Y., Zhang, W., Yang, Y. 2022. A comprehensive benchmark analysis for sand dust image reconstruction. Journal of Visual Communication and Image Representation, 89, 103638.

SSBE 2024: PYTHON-BASED OPEN SOFTWARE FOR EEG-BASED BRAIN-MACHINE INTERFACES

Omar Lone*, Manuel Weiss, Daniel Baumgartner, Ricardo Chavarriaga

1. ZHAW School of Engineering, Switzerland

*Corresponding author: lone@zhaw.ch

Introduction

The growing desire to seamlessly and intuitively interact with technology has driven significant advancements in two key areas, namely Brain-Machine Interfaces (BMIs) and Human-Machine Interfaces (HMIs). BMIs represent the establishment of a direct link between the brain and external devices. Translating thought into action by decoding brain activity captured with sensors like EEG, MEG or fMRI. HMIs, on the other hand, encompass a broader range of technologies that facilitate information exchange between humans and machines. The type of interaction ranges from conventional keyboards and mice to complex interfaces as advanced virtual reality systems with haptic feedback. As these fields continue to evolve, the convergence of human and machine intelligence has the potential to expand various scientific domains, including neurorehabilitation, prosthetic control, and communication for individuals with disabilities. Electroencephalography (EEG) holds promise for brain-computer interfaces (BCIs), but current setups limit real-world use. Traditional EEG requires controlled labs, and specialized software, hindering portability and ease of use. Existing software tools offer some solutions but fall short. GUI-based interfaces like OpenVibe, while offering closed-loop support, lack flexibility for integration. Conversely, powerful Python libraries like EEG-ExPy struggle with real-time processing needed for closed-loop experiments [1], [2], [3].

Material and Methods

This project presents a modular Python framework for real-time EEG processing, see Figure 1. Built on open-source tools, it offers data acquisition, pre-processing, feature extraction, classification, and real-time visualization, as illustrated in Figure 2. This facilitates the development and evaluation of processing strategies for BCI applications. It is user-friendly for programmers and data scientists, enables the usage of all commonly used Python modules, is configurable, and allows for an easier interdisciplinary workflow on BCI systems and provides benchmark for different acquisition methods, see Table 1.

Table 1: Acquisition Benchmark

API	Get Data Time per call [ms]
LSL	3.98
Unicorn C	3.97
Unicorn Python	3.88
BrainFlow	0.09



Annual Meeting of the Swiss Society of Biomedical Engineering SSBE
Zurich University of Applied Sciences ZHAW
IMES Institute of Mechanical Systems, Winterthur
September 5, 2024

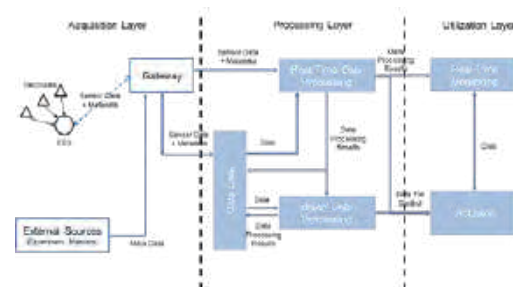


Figure 1: Illustration of the framework



Figure 2: Signal Visualization

References

- [1] R. K. Das, A. Martin, T. Zuraes, D. Dowling, and A. Khan, "A Survey on EEG Data Analysis Software," *Sci*, vol. 5, no. 2, Art. no. 2, Jun. 2023, doi: 10.3390/sci5020023.
- [2] H. Anowar, "A Python-based Brain-Computer Interface Package for Neural Data Analysis".
- [3] M. A. Lebedev and M. A. L. Nicolelis, "Brain-Machine Interfaces: From Basic Science to Neuroprostheses and Neurorehabilitation," *Physiol. Rev.*, vol. 97, no. 2, pp. 767–837, Apr. 2017, doi: 10.1152/physrev.00027.2016.

Evaluating dry EEG technology out of the lab

Mushfika Sultana <i>BCI-NE, CSEE</i> <i>University of Essex</i> Colchester, UK ms17811@essex.ac.uk	Osheen Jain <i>Computing, Psychology Dept.</i> <i>Goldsmiths, UoL</i> London, UK ojain001@gold.ac.uk	Sebastian Halder <i>BCI-NE, CSEE</i> <i>University of Essex</i> Colchester, UK 0000-0003-1017-3696	Ana Matran-Fernandez <i>BCI-NE, CSEE</i> <i>University of Essex</i> Colchester, UK 0000-0002-8409-3747	Rab Nawaz <i>BCI-NE, CSEE</i> <i>University of Essex</i> Colchester, UK rab.nawaz@essex.ac.uk
Reinhold Scherer <i>BCI-NE, CSEE</i> <i>University of Essex</i> Colchester, UK 0000-0003-3407-9709	Ricardo Chavarriaga <i>Responsible AI Innovation Group</i> <i>ZHAW School of Engineering</i> Winterthur, Switzerland 0000-0002-8879-2860	José del R. Millán <i>Carol Cockrell Curran Endowed Chair</i> <i>University of Texas at Austin</i> Austin, USA 0000-0001-5819-1522	Serafeim Perdikis <i>BCI-NE, CSEE</i> <i>University of Essex</i> Colchester, UK 0000-0003-2033-2486	

Abstract—Dry electroencephalography (EEG) electrodes have emerged as a promising solution for enhancing the usability of non-invasive Brain-Computer Interface (BCI) systems. Recent advancements in dry helmets have demonstrated competitiveness in enabling EEG and BCI applications with state-of-the-art gel-based systems. In this study, we evaluate the performance and signal quality of a dry EEG cap for BCI applications in a large population. The assessment was conducted under extremely noisy conditions at a public exhibition, surrounded by numerous attendees and multiple sources of electromagnetic interference. Our analysis includes 100 participants from the Mental Work exhibition and assesses the acquired signal's integrity through, Motor Imagery (MI) classification accuracy, the kurtosis of the raw signals and the deviation of the Power Spectral Density (PSD) spectra from the ideal curve. Results show that 56 participants achieved above chance-level single-sample accuracy, with even higher accuracy observed for peak performance. Frontal channels exhibited higher artifact presence, yet the overall signal quality was comparable to gel-based systems used in the authors' previous studies. These findings confirm the potential of dry EEG technology to enable BCI applications beyond the lab, facilitating everyday use in homes, clinics, and public spaces.

Index Terms—EEG, dry sensors, brain-computer interface, evaluation

I. INTRODUCTION

Non-invasive EEG brain imaging remains the most popular choice for BCI applications to date. However, state-of-the-art "wet" EEG sensors, which use non-abrasive gel to enhance conductivity, are often criticized for their obtrusiveness and logistical burden. While these sensors provide the highest signal quality in terms of Signal-to-Noise Ratio (SNR), these drawbacks significantly limit their practical application [1].

The emergence of various dry EEG sensors over the past 15 years has not only aimed to overcome the limitations of gel-based systems but has also facilitated the expansion of EEG applications beyond laboratory environments into more practical, real-world scenarios. The need for portable and easy-to-use EEG systems is driven by multiple factors, including the increasing interest in BCIs, neurofeedback applications, and health monitoring. Dry EEG systems, which use sensors

that do not require conductive gels or saline, offer a promising solution by reducing setup time and improving user comfort. The potential for these systems to facilitate long-term monitoring and real-time brain activity analysis in naturalistic settings is particularly appealing for both clinical and research applications. Initial studies on dry EEG highlighted challenges like lower signal quality, higher vulnerability to motion artifacts and user intolerance to long-term use compared to their wet counterparts [2]. Despite these challenges, advances in sensor materials, signal processing algorithms and mechanical design have significantly improved the performance and comfort of dry EEG systems. Researchers have developed novel electrode designs and adaptive filtering techniques to address issues related to skin-electrode impedance and signal noise.

Numerous studies have demonstrated that while dry EEG systems may exhibit higher impedance and lower signal-to-noise ratios, they can still capture clinically relevant brain activity patterns. Hinrichs et al. [1] conducted a comparative study showing that dry EEG systems could reliably detect Event-Related Potential (ERP) responses and other standard EEG markers. In the study reported in [3], a novel EEG system using dry electrodes was compared to a clinically available EEG system employing wet electrodes. Similarly, Guger et al. [4] successfully validated a dry electrode BCI system for P300-based spelling, achieving comparable accuracy to gel-based systems while ensuring user comfort, thereby demonstrating its potential for practical and convenient use in real-world settings. Other researchers proposed a new dry EEG system comparing it to traditional wet electrodes in EEG recordings for basic research and BCI [5], finding that dry electrodes can reliably detect EEG signals and suggesting their potential for broader BCI accessibility outside of lab settings. Liao et al. [2] and Yeung et al. [6] also conducted comparisons of dry and wet EEG sensors, focusing on foam-based sensors and spring-loaded electrodes using linear correlation analyses. The ensemble of these works suggests that, despite some persisting limitations, dry EEG technology is maturing and holds considerable promise for imminent applicability in real-

world BCI scenarios.

One of the most significant advantages of dry EEG technology is its potential for use in situations requiring everyday, long-duration BCI operation. This capability opens up new possibilities for continuous monitoring of brain activity in individuals with neurological conditions, such as epilepsy [7], [8], where real-time detection of abnormal brain activity is crucial for timely intervention. Moreover, the use of dry EEG in non-clinical settings, such as attention training programs for treating Attention Deficit Hyperactivity Disorder (ADHD) [9], real-time driving fatigue detection system [10] and finding biomarkers of mental workload [11] can provide valuable insights that were previously impractical with traditional lab-based setups.

As elaborated above, many recent studies have provided solid scientific evidence suggesting that some of the current, commercially available dry EEG caps are capable of delivering comparable performance to that obtained with wet systems [1]–[3], [5], [6], [12]–[14]. Nevertheless, most of these investigations reported findings on relatively small subject samples, either in fully controlled laboratory conditions or only minimally cluttered environments, and often without estimating BCI or other types of application performance. On the other hand, considerable efforts have been devoted to evaluating state-of-the-art paradigms—such as Sensori Motor Rhythms (SMR), P300s, and Steady-state visually evoked potentials (SSVEPs)—using gel-based systems. Those evaluations aim to determine the proportion of the population that can effectively utilize each type of BCI and to explore ways to enhance BCI usability [15]. This study investigates the potential to operate a standard SMR BCI paradigm through a state-of-the-art dry EEG cap, with a sizeable sample of 100 participants extracted for preliminary analysis out of an unprecedentedly large database hosting EEG data by 530 volunteers. Importantly, EEG recording took place in the framework of a crowded, public exhibition and in the presence of numerous nearby electromagnetic interference sources. Our analysis indicates that dry technology has considerably advanced and may be ready to enable unobtrusive, plug-and-play BCI applications even in extremely challenging, real-world settings.

II. METHODS

A. Participants

We analyzed data of the first 100 registered out of the 530 total participants of the Mental Work event [17]. At Mental Work, a public, open-door exhibition that took place across several months in 2017 and 2018 at the premises of École Polytechnique Fédérale de Lausanne (EPFL), Switzerland, the public used research-grade dry EEG helmets to connect to several different robotic machines/sculptures inspired by the industrial revolution, and activate them with a binary (2-class) MI BCI developed by EPFL researchers. Participants were visitors of the exhibition who registered online and booked a Mental Work session.

B. Experimental setups and protocols

Each session comprised a 30-minute training/calibration session to collect EEG data for supervised estimation of a machine learning decoder and to briefly train the subject in performing mental imagery, followed by cued, closed-loop control of a conventional “feedback bar” graphical application on a screen; subsequently, participants would proceed with free-control of each of the three available robots through MI. The training and robot control was conducted in highly noisy conditions, surrounded by over 100 exhibition attendees. The environment was rife with noise and interference from cell phones, Bluetooth devices, and various other sources of electromagnetic disruption. Calibration took place in a non-isolated booth within the exhibition’s space, accessible to observers and vulnerable to auditory and electromagnetic noise. It involved 3 training runs yielding a total of 30, 4-s long trials for each MI class. The two mental tasks consisted of kinaesthetic imagination of right and left-hand movements. Here, we only report the calibration data of each user. A DSI-24 wireless, portable, dry, EEG cap (Wearable Sensing, San Diego, CA, USA) with on-board artifact elimination was used to acquire the signal from 19 EEG channels (10-20 system locations: Fp1, Fp2, Fz, F3, F4, F7, F8, Cz, C3, C4, C5, C6, T7, T8, Pz, P3, P4, O1, O2, two earlobe clip references, ground electrode at FPz) at 300 Hz sampling rate.

C. Data Analysis and Evaluation

EEG signals are processed within each trial with linear detrending, DC removal, second-order Butterworth band-pass filtering with low/high cut-off frequencies at 2 Hz and 40 Hz, respectively, and large Laplacian spatial filtering. We extracted PSD features from all 19 EEG channels with 0.5 Hz frequency resolution between [0.5, 30] Hz, in sliding 1 s windows with 0.5 s overlapping/shift using the Welch method (internal windows 0.5 s long with 50% overlapping). This yields 420 samples (60 trials $\times$ 7 samples/trial, 240 per class) and 1140 candidate features (60 bands $\times$ 19 channels) for each subject. Classification accuracy is estimated with 5-fold cross-validation individually for each subject and run using a Quadratic Discriminant Analysis (QDA) classifier, selecting within each iteration the 10 best features in terms of r^2 separability from a candidate feature subset consisting of channels Cz, C3, C4, C5, C6 and the μ (8–14 Hz) and β (18–24 Hz) bands, known to be relevant to SMR BCI. Samples from the same trial are kept in the same cross-validation fold to avoid overestimating accuracy due to data dependence. To evaluate subject-wise performance, we report both the mean and the maximum accuracy across the subject’s three runs. We compare each subject’s MI BCI performance with the 58% chance level, which is based on the assumption that classification decisions follow a binomial distribution. This 58% threshold corresponds to the 95% confidence interval for non-random classification accuracy in a binary classification problem, given the amount of data available per subject [16].

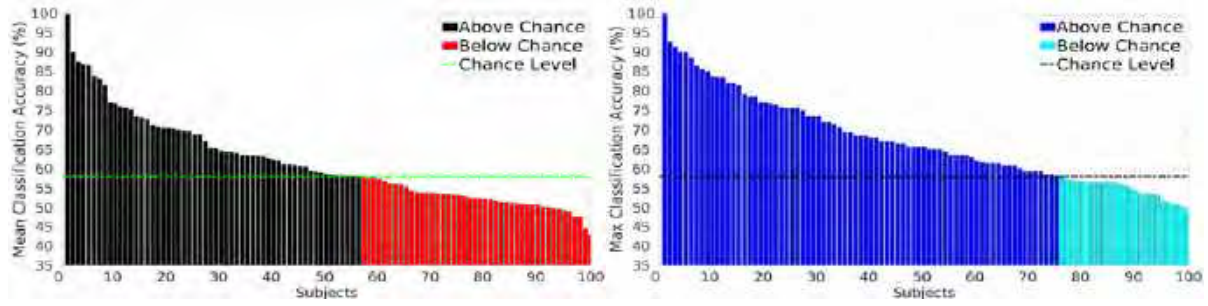


Fig. 1. Sorted mean (left) and maximum (right) classification accuracy for all subjects. The dashed (green and black) lines indicate the 95% confidence interval for non-random classification in a binary classification problem with the amount of data available per subject here [16].

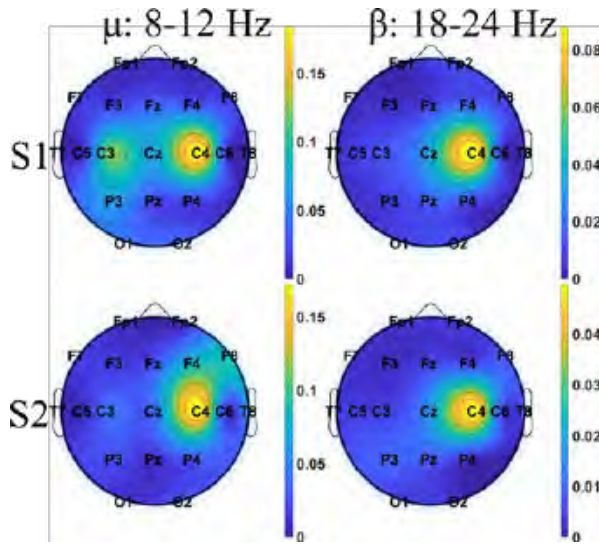


Fig. 2. Topographic distribution of r^2 feature separability averaged within the μ (left) and the β band (right) for the best (S1, top) and second-best (S2, bottom) subject in terms of classification accuracy.

D. Quality assessment of recorded EEG data

EEG data are often contaminated by various artifacts that mask true EEG signals. These artifacts can be divided into physiological (e.g., ocular, muscle, cardiac, motion) [18]–[20] and non-physiological (i.e., electromagnetic interference, acquisition system malfunctions, physical tampering) [21]. Physiological artifacts arise from involuntary physiological activities, while non-physiological artifacts stem from external influences [19]. Artifacts reduce the signal-to-noise ratio (SNR) and interfere with analysis, making artifact detection and removal essential for restoring the quality of EEG signals. High-quality acquisition systems and standardized experimental procedures can mitigate these effects, though some impact of artifacts remains inevitable in most situations. In particular, EEG signals recorded outside the lab and in real-world environments are prone to contain many artifacts. Artifacts

substantially differ from clean EEG signals in various properties. To assess the quality of the recorded signal using dry EEG technology, we hereby focus on the following metrics:

- The kurtosis value of the signal, implicitly measures the tendency of observing outliers of the signal amplitude over time [22], [23].
- The dissimilarity of the PSD spectrum of the observed EEG to a reference distribution [24].

Kurtosis k is the fourth standardized moment of a real-valued random variable defined as $k = m_4/s^4$, where $m_4 = \mathbb{E}[(X - m)^4]$ the variable's fourth central moment and s its standard deviation. It offers insights into the amplitude distribution of the signal over time, helping to identify deviations that indicate the presence of artifacts. Specifically, it evaluates whether the data distribution is more peaked or flat relative to the normal distribution [22]. Abnormal EEG amplitude is determined by setting a threshold k_{th} on the kurtosis calculated within 1 s windows. We examine three variants of thresholds k_{th} —statistical [25], percentile-based [26] and robust statistical-based (particularly the Median and Median Absolute Deviation (MAD)) [27].

For the statistical approach, the threshold is chosen to be $k_{th} = \mu(k) + c \times \sigma(k)$, where $\mu(k)$ the mean of the kurtosis values over all channels and time points, $\sigma(k)$ the corresponding standard deviation, and c a hyperparameter controlling the conservativeness of this metric. The statistical thresholding approach is reported for values $c = 0.5, 1, 2$ in accordance with [25]. Of note, increasing c accounts for a less conservative prognosis of artifactual data (i.e., large k values may be considered normal). This method highlights channels with kurtosis significantly higher than the average. For the percentile-based approach, a threshold is set at the 95th percentile of k values. This method identifies the top 5% of k values, ensuring that only the segments with the most extreme kurtosis are flagged. The robust statistical approach is less sensitive to outliers, using the median and MAD to set thresholds. The threshold is calculated as $k_{th} = \text{median}(k) + k_{robust} \cdot \text{mad}(k)$, with $k_{robust} = 3$. This method ensures that even slight deviations in kurtosis are accounted for without being overly influenced by extreme outliers.

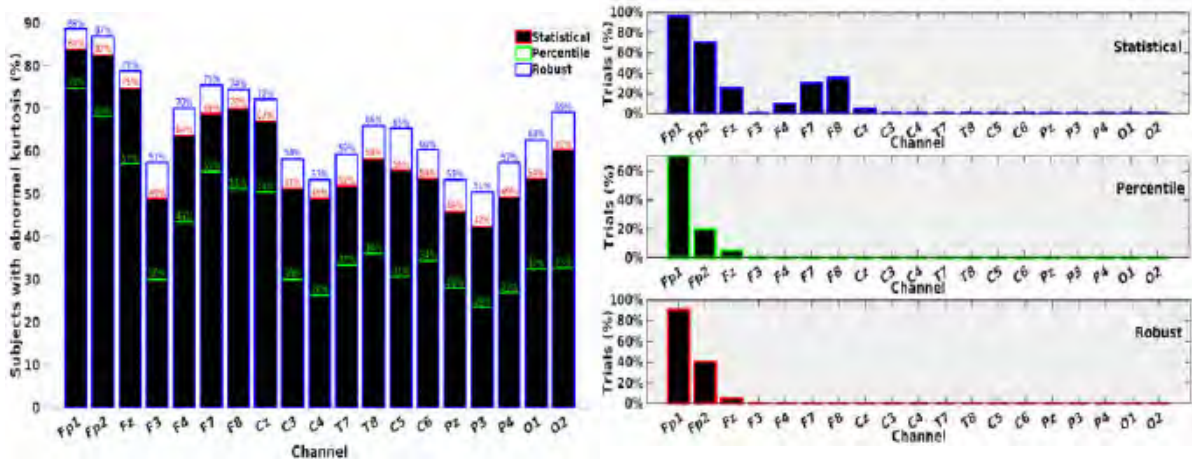


Fig. 3. Left: Percentage of subjects with abnormal average kurtosis value across EEG channels for different k_{th} thresholds. Right: Percentage of trials with abnormal average kurtosis value across EEG channels for the statistical (top, $c = 1.0$), percentile-based (middle) and robust statistical (bottom) k_{th} threshold. Thresholds for the subject-wise percentages are computed on the sample of $N = 100$ subject-averaged kurtosis values. Trial-wise percentages are computed on the sample of $N = 100 = 6000$ trial-averaged kurtosis values, pulling together all trials of all subjects in the dataset.

Apart from kurtosis, the PSD spectrum dissimilarity measure compares the observed EEG signal's frequency content to a reference distribution, detecting discrepancies known to be caused by artifacts. The power spectra of clean EEG signals generally follow a $1/f$ curve [24]. Notwithstanding the fact that certain cognitive tasks may cause deviations from this inverse-proportionality rule in particular frequency bands (such as the α and β bands during MI [25], [28]), in most cases, large deviations (especially those that affect broad band ranges) are indicative of artifact-contaminated EEG. For each channel separately, we compute a dissimilarity index DS_{PSD} as the Euclidean distance between the PSD spectrum over $[0 : 0.5 : 30]$ Hz extracted in consecutive 1s windows and the corresponding ideal $1/f$ curve; the S_{PSD} values over a certain threshold are marked as abnormal. Results are reported for thresholds 1.5 and 3.5 as in [25].

III. RESULTS

Fig. 1 illustrates the classification accuracy results for 100 subjects using the dry EEG cap after only 30 minutes of open-loop training under difficult circumstances. The left panel shows the mean classification accuracy, where 56 out of 100 subjects achieved above-chance accuracy, with 44 subjects falling below the chance level (indicated in red). Notably, five participants obtained accuracy exceeding 85% (including 100% exhibited by one participant). The right side panel depicts the maximum classification accuracy (best of three runs), where 75 subjects reached above-chance accuracy (in blue), while 25 subjects fell below the chance level (in cyan). It must be highlighted that the presented single-sample accuracy estimate refers to samples derived from 1s data segments, as opposed to the single-trial estimates usually reported in the literature that reflect information from 4–5 s of EEG data.

Sample-wise classification is the norm for most actual, closed-loop MI BCI applications [29] where the BCI must be sensitive enough to allow subjects to switch among different BCI commands very fast, as needed. Previous experience indicates that the most commonly reported trial-based accuracy will tend to be 10-15% higher than the sample-based equivalent; consequently, the chance level threshold shown here approximately corresponds to the 70% trial-based accuracy threshold considered to be adequate for BCI applications, as a rule of thumb [30]. Effectively, these results imply that the evaluated dry EEG system would be adequate to support MI BCI control under realistic conditions for 55-75% of the population.

Fig. 2 confirms that high classification accuracy is fueled by the anticipated SMR patterns that can be precisely captured by the dry system used. Specifically, contralateral activation associated with MI of both hands is observed for S1 in the μ band, and for the left hand in the β (S1, S2) and μ band (S2). SMR modulation is stronger in the μ band as is most often the case for able-bodied users.

Fig. 3 presents the percentage of subjects with suspiciously high kurtosis k values across different EEG channels, using all three thresholding criteria we have introduced. Channels Fp1, Fp2, and Fz seem to be those mostly affected for the vast majority of subjects. Conversely, channels like F3, C4, C3, and Pz exhibit lower percentages, indicating cleaner signals in these regions. Overall, the majority of subjects seem to enjoy good quality signals in those EEG channels that are not very close to the head's periphery, especially when considering the less conservative thresholding approaches; of note, these are also the channels involved in known EEG correlates for the most common BCI paradigms. The right panel of Fig. 3 confirms this finding, by illustrating that only the very frontal EEG channels, namely channels Fp1, Fp2 and

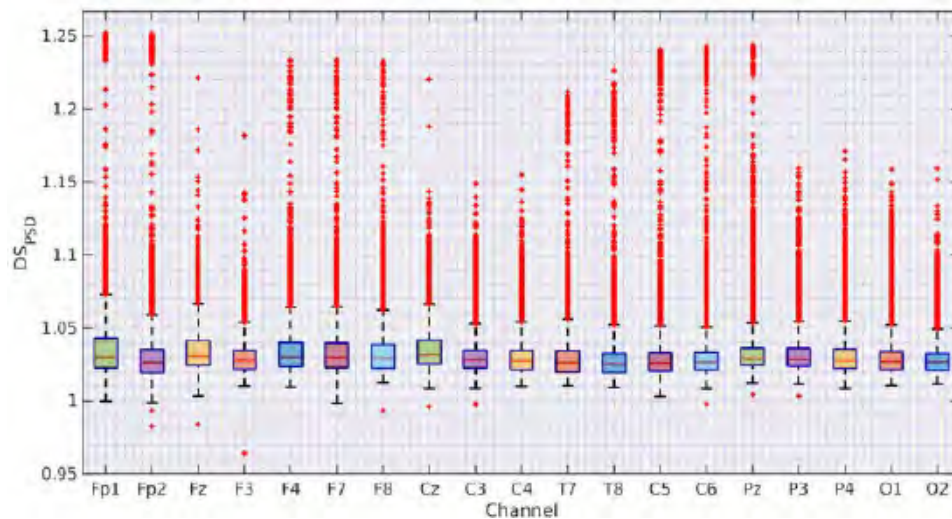


Fig. 4. Average DS_{PSD} dissimilarity per EEG channel pulling all subjects together. The box plots show the distribution of DS_{PSD} per channel (median, 25th and 75th percentile). Red markers indicate outliers.

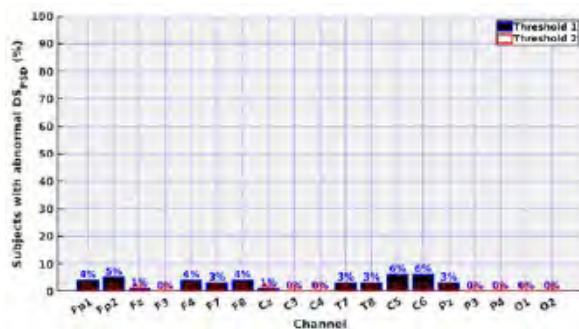


Fig. 5. Percentage of subjects with abnormal average DS_{PSD} across EEG channels using two different thresholds: Threshold 1 corresponds to $DS_{PSD} > 1.2$ and Threshold 2 to $DS_{PSD} > 3.5$ [25].

Fz exhibit highly irregular kurtosis levels for large percentages of trials. On the contrary, it is positively surprising that central, parietal and even occipital channels (e.g., Cz, C3, C4, Pz, O1, O2) seem to be barely affected. Importantly, the different thresholding methodologies, irrespective of the fact that each corresponds to a different level of conservativeness, are shown to be consistent with one another. It must be underlined that subject-wise percentages are far more conservative because all the defined distribution-based thresholds are calculated on the smaller sample of 100 subjects, compared to a sample of 6,000 trials for the respective trial-wise results. Taken together, these effects are very similar to those anticipated for a state-of-the-art wet EEG system, as frontal EEG channels are likely to be influenced by the very common, frequently occurring and very strong (at least two orders of magnitude bigger than pure EEG) ocular and facial muscle artifacts independent of the

EEG system's quality.

Fig. 4 illustrates the DS_{PSD} dissimilarity index across all EEG channels of the used layout for all subjects. Again, frontal channels Fp1, Fp2, and Fz, but also other peripheral channels such as C5 and C6, show the highest median distance from the ideal $1/f$ spectrum curve, indicating the potential impact of artifacts on these channels. Conversely, channels such as F3, F4, F7, F8, and Cz exhibit moderate DS_{PSD} . These deviations are less pronounced compared to the frontal channels but are still noteworthy. Furthermore, central, parietal, and occipital channels (e.g., C3, C4, Pz, O1, O2) generally show lower median DS_{PSD} . More than the median value, the presence (magnitude and number) of outliers is a clearer sign that frontal channels exhibit the highest irregularity.

Fig. 5 shows the percentage of subjects (per channel) showing high DS_{PSD} values, using two different thresholds, 1.2 and 3.5 as in [25]. For the first threshold, the results are fully in line with those of Fig. 4. The second threshold proposed in the relevant literature proves to be too tolerant, with no channel flagged as artifact-inflicted.

IV. DISCUSSION

This preliminary analysis of the Mental Work exhibition dataset demonstrates that state-of-the-art dry EEG caps hold great promise for enabling brain control in realistic application scenarios, while significantly enhancing the ease of use, which is the Achilles' heel of current BCI prototypes. Overall, 56 participants attained a level of (open-loop) MI BCI aptitude exceeding the 58% chance classification level. Notably, even higher accuracy levels were observed when considering maximum accuracy, indicating that accuracy results may improve with user training, as shown in research conducted with wet systems [29]. The reported kurtosis values indicate the

soundness of signal quality for the majority of EEG locations, except for the frontal channels, which are typically prone to artifacts also in gel-based EEG systems. The reported dissimilarity of the EEG electrode spectra to the ideal curve further corroborates that only frontal and peripheral regions may exhibit low signal quality. These findings align well with similar analyses conducted on gel-based EEG systems, as well as with studies comparing other commercial dry EEG helmets with wet electrodes.

Our results indicate that modern dry EEG technology has considerably reduced the signal quality gap with gel-based systems. However, this claim needs to be better substantiated through explicit comparisons with such systems. Our future work will comprise such comparisons and will additionally focus on extracting several additional measures of signal quality, such as temporal dependency and auto-mutual information, spiking activity analysis, and other measures that have been developed in recent years [25]). Last but not least, we plan to process the entirety of the immense Mental Work dataset, thus presenting the most well-powered comparative analysis of EEG system typologies reported to date.

REFERENCES

- [1] H. Hinrichs, M. Scholz, A. K. Baum, J. W. Y. Kam, R. T. Knight, and H.-J. Heinze, "Comparison between a wireless dry electrode eeg system with a conventional wired wet electrode eeg system for clinical applications," *Scientific Reports*, vol. 10, no. 1, Mar. 2020. [Online]. Available: <http://dx.doi.org/10.1038/s41598-020-62154-0>
- [2] I. J. W. S. F. C. S. Y. L. B. W. C. J. Y. C. C. T. L. Lun De Liao, Chi Yu Chen, "Gaming control using a wearable and wireless eeg-based brain-computer interface device with novel dry foam-based sensors," *Journal of NeuroEngineering and Rehabilitation*, vol. 9, no. 1, pp. 1–12, 2012. [Online]. Available: <https://jneuroengrehab.biomedcentral.com/articles/10.1186/1743-0003-9-5>
- [3] P. B. M. C. C. J. A. M. A. v. S. Gaetano Gargiulo, Rafael A. Calvo, "A new eeg recording system for passive dry electrodes," *Clinical neurophysiology : official journal of the International Federation of Clinical Neurophysiology*, vol. 121, no. 5, pp. 686–693, 2010. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/20097606/>
- [4] B. Z. A. G. E. Christoph Guger, Gunther Krausz, "Comparison of dry and gel based electrodes for p300 brain-computer interfaces," *Frontiers in Neuroscience*, vol. 6, no. MAY, 2012. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3345570/>
- [5] R. E. S. D. Maarten De Vos, Markus Kroesen, "P300 speller bci with a mobile eeg system: comparison to a traditional amplifier," *Journal of neural engineering*, vol. 11, no. 3, pp. 1–12, 2014. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/24763067/>
- [6] C. V. T. P. M. O. B. S. M. N. V.-B. Arnold Yeung, Harinath Garudadri, "Comparison of foam-based and spring-loaded dry eeg electrodes with wet electrodes in resting and moving conditions," *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference*, vol. 2015, pp. 7131–7134, 2015. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/26737936/>
- [7] J. I. Doerrfuss, T. Kilic, M. Ahmadi, J. E. Weber, and M. Holtkamp, "Predictive value of acute eeg measurements for seizures and epilepsy after stroke using a dry cap electrode eeg system—study design and proof of concept," *Epilepsy & Behavior*, vol. 104, p. 106486, 2020.
- [8] F. Pinho, J. Cerqueira, J. Correia, N. Sousa, and N. Dias, "mybrain: a novel eeg embedded system for epilepsy monitoring," *Journal of medical engineering & technology*, vol. 41, no. 7, pp. 564–585, 2017.
- [9] C. G. Lim, T. S. Lee, C. Guan, D. S. S. Fung, Y. Zhao, S. S. W. Teng, H. Zhang, and K. R. R. Krishnan, "A brain-computer interface based attention training program for treating attention deficit hyperactivity disorder," *PloS one*, vol. 7, no. 10, p. e46692, 2012.
- [10] H. Wang, A. Dragomir, N. I. Abbasi, J. Li, N. V. Thakor, and A. Bez-erianos, "A novel real-time driving fatigue detection system based on wireless dry eeg," *Cognitive neurodynamics*, vol. 12, no. 4, pp. 365–376, 2018.
- [11] W. K. So, S. W. Wong, J. N. Mak, and R. H. Chan, "An evaluation of mental workload with frontal eeg," *PloS one*, vol. 12, no. 4, p. e0174949, 2017.
- [12] J. W. Kam, S. Griffin, A. Shen, S. Patel, H. Hinrichs, H.-J. Heinze, L. Y. Deouell, and R. T. Knight, "Systematic comparison between a wireless eeg system with dry electrodes and a wired eeg system with wet electrodes," *NeuroImage*, vol. 184, pp. 119–129, 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1053811918307961>
- [13] C. R. Ng, P. Fiedler, L. Kuhlmann, D. Liley, B. Vasconcelos, C. Fonseca, G. Tamburro, S. Comani, T. K.-Y. Lui, C.-Y. Tse, I. F. Warsito, E. Supriyanto, and J. Hauelsen, "Multi-center evaluation of gel-based and dry multipin eeg caps," *Sensors*, vol. 22, no. 20, 2022. [Online]. Available: <https://www.mdpi.com/1424-8220/22/20/8079>
- [14] T. Neumann, A. K. Baum, U. Baum, R. Deike, H. Feistner, M. Scholz, H. Hinrichs, and B.-P. Robra, "Assessment of the technical usability and efficacy of a new portable dry-electrode eeg recorder: First results of the homeone study," *Clinical Neurophysiology*, vol. 130, no. 11, pp. 2076–2087, 2019.
- [15] G. Edlinger, B. Z. Allison, and C. Guger, "How many people can use a bci system?" *Clinical systems neuroscience*, pp. 33–66, 2015.
- [16] G. Müller-Putz, R. Scherer, C. Brunner, R. Leeb, and G. Pfurtscheller, "Better than random? A closer look on BCI results," *International Journal of Bioelectromagnetism*, vol. 10, no. 1, pp. 52–55, 2008.
- [17] "Mental Work: the cognitive revolution starts here," <https://mentalwork.net/project/>.
- [18] R. J. Croft and R. J. Barry, "Removal of ocular artifact from the eeg: a review," *Neurophysiologie Clinique/Clinical Neurophysiology*, vol. 30, no. 1, pp. 5–19, 2000.
- [19] M. K. Islam, A. Rastegamia, and Z. Yang, "Methods for artifact detection and removal from scalp eeg: A review," *Neurophysiologie Clinique/Clinical Neurophysiology*, vol. 46, no. 4-5, pp. 287–305, 2016.
- [20] A. Schlögl, C. Keinrath, D. Zimmermann, R. Scherer, R. Leeb, and G. Pfurtscheller, "A fully automated correction method of eeg artifacts in eeg recordings," *Clinical neurophysiology*, vol. 118, no. 1, pp. 98–104, 2007.
- [21] X. Jiang, G.-B. Bian, and Z. Tian, "Removal of artifacts from eeg signals: a review," *Sensors*, vol. 19, no. 5, p. 987, 2019.
- [22] D. Mantini, M. G. Perrucci, C. Del Gratta, G. L. Romani, and M. Corbetta, "Electrophysiological signatures of resting state networks in the human brain," *Proceedings of the National Academy of Sciences*, vol. 104, no. 32, pp. 13170–13175, 2007.
- [23] B. Kamousi, A. M. Grant, B. Bachelder, J. Yi, M. Hajinoroozi, and R. Woo, "Comparing the quality of signals recorded with a rapid response eeg and conventional clinical eeg systems," *Clinical Neurophysiology Practice*, vol. 4, pp. 69–75, 2019.
- [24] L. M. Ward, *Dynamical cognitive science*. MIT press, 2002.
- [25] I. Daly, R. Scherer, M. Billinger, and G. Müller-Putz, "Force: Fully online and automated artifact removal for brain-computer interfacing," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 23, no. 5, pp. 725–736, 2014.
- [26] R. Mahajan and B. I. Morshed, "Unsupervised eye blink artifact de-noising of eeg data with modified multiscale sample entropy, kurtosis, and wavelet-ica," *IEEE journal of Biomedical and Health Informatics*, vol. 19, no. 1, pp. 158–165, 2014.
- [27] X. Yong, R. K. Ward, and G. E. Birch, "Robust common spatial patterns for eeg signal preprocessing," in *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE*, 2008, pp. 2087–2090.
- [28] G. Pfurtscheller and F. L. Da Silva, "Event-related eeg/meg synchronization and desynchronization: basic principles," *Clinical neurophysiology*, vol. 110, no. 11, pp. 1842–1857, 1999.
- [29] S. Perdakis, L. Tonin, S. Saeedi, C. Schneider, and J. d. R. Millán, "The Cyathlon BCI race: Successful longitudinal mutual learning with two tetraplegic users," *PLoS Biol.*, vol. 16, no. 5, p. e2003787, 2018.
- [30] A. Kübler, N. Neumann, B. Wilhelm, T. Hinterberger, and N. Birbaumer, "Predictability of brain-computer communication," *J. Psychophysiol.*, vol. 18, pp. 121–129, 2004. [Online]. Available: <https://doi.org/10.1027/0269-8803.18.23.121>

Zurich University
of Applied Sciences

certAlnty – A Certification Scheme for AI Trustworthiness An Innosuisse Project

Centre for Artificial Intelligence (CAI)¹
Institute of Applied Mathematics and Physics (IAMP)²
CertX AG

Philipp Denzel¹, Oliver Forster¹, Yann Billeter¹, Frank-Peter Schilling¹, Ricardo Chavarriaga¹,
Carmen Mei-Ling Frischknecht-Gruber², Stefan Brunner², Monika Reif², Joanna Weng²

Background

The European Union's AI Act, effective August 1, 2024, establishes a comprehensive regulatory framework for AI systems. As organisations and certifiers prepare to demonstrate compliance, practical guidelines for achieving AI trustworthiness remain limited, both within the EU and globally. The AI Act categorises AI systems by risk – **unacceptable, high, limited, and minimal** – while unacceptable-risk practices, such as biometric classification, emotion recognition, subliminal manipulation, and social scoring, are prohibited. High-risk systems, particularly in sectors like healthcare, transport, education, and energy, must meet strict requirements for market entry, including CE-marking certification.

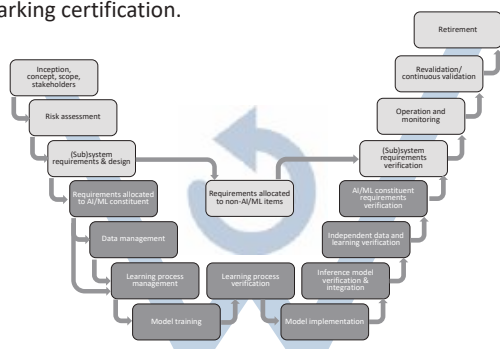


Figure 1 – The AI-based system lifecycle (source: EASA adapted)

High-risk AI systems also undergo conformity assessments and third-party certification to ensure transparency, human oversight, fairness, reliability and safety and security, as well as the protection of fundamental rights. Meanwhile, limited-risk AI systems are obligated to maintain the key aspect of transparency regarding their purpose, function, and decision-making processes. Supporting these regulations are standards from organisations such as ISO/IEC, IEEE, and NIST. ISO/IEC standards address AI terminology, performance metrics, data quality, and ethics. Whereas the IEEE P7000 series focuses on the ethical implications of AI technologies, while NIST provides frameworks for risk management, data quality, and transparency, offering essential guidance for demonstrating compliance and promoting best practices.

CertAlnty Project Overview

The CertAlnty project has developed a comprehensive Certification Scheme for AI Systems to ensure their trustworthiness. The scheme defines clear objectives and corresponding means of compliance, criteria, and evaluation measures aligned with relevant standards and regulations. It offers technical and scientific methods for assessing key AI properties, such as transparency, human oversight, data governance, reliability, fairness, and safety and security.

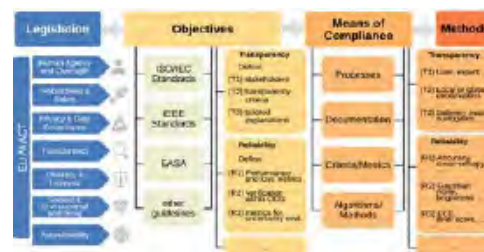


Figure 2 – Certification scheme for AI systems. It enables identification of application-specific requirements, respective means of compliance and technical methods for conformity assessment

These methods support systematic evaluations throughout the AI lifecycle – from risk assessment, requirements and data acquisition to development, testing, and deployment. Based on an analysis of 38 standards (ISO/IEC, IEEE, EASA) and integration of EU legislation, the scheme bridges regulatory requirements with technical assessments. It identifies 95 techniques for measuring attributes like explainability, robustness, and security, ensuring compliance with the EU AI Act.

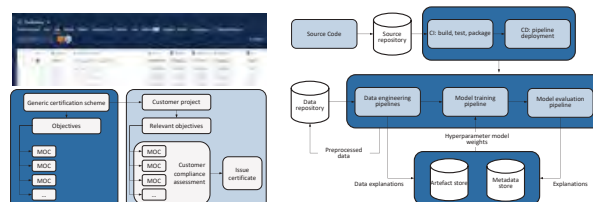


Figure 3 – Practical application: On the left a visualisation of the certification workflow based on JIRA and on the right an overview of the MLOps system architecture.

To ensure a reliable and practical approach to AI certification, the CertAlnty project uses Jira for requirements management. This tool ensures traceability, systematic documentation and real-time collaboration in line with standards such as ISO 9001, ISO/IEC 27001 and ISO/IEC 23894. Jira's features including version control, decision tracking and customisable dashboards improve efficiency and consistency throughout the certification process. Complementary MLOps practices ensure trustworthiness by design, incorporating best practices at every stage of the AI lifecycle. Our MLOps infrastructure supports model development, training, continuous integration and continuous deployment and monitoring, using tools such as GitHub, MLflow and Apache Airflow. This setup maintains traceability, automates testing and ensures ongoing standards compliance. The workflow has been validated in three high-risk use cases - medical applications, autonomous driving and vehicle detection at construction sites - aligning with the requirements of the EU AI Act and demonstrating effective certification processes for different data types and compliance requirements.

AI Assessment in Practice: Implementing a Certification Scheme for AI Trustworthiness

Carmen Frischknecht-Gruber ✉ 

Zurich University of Applied Sciences ZHAW,
Winterthur, Switzerland

Philipp Denzel ✉ 

Zurich University of Applied Sciences ZHAW,
Winterthur, Switzerland

Monika Reif ✉ 

Zurich University of Applied Sciences ZHAW,
Winterthur, Switzerland

Yann Billeter ✉ 

Zurich University of Applied Sciences ZHAW,
Winterthur, Switzerland

Stefan Brunner ✉

Zurich University of Applied Sciences ZHAW,
Winterthur, Switzerland

Oliver Forster ✉ 

Zurich University of Applied Sciences ZHAW,
Winterthur, Switzerland

Frank-Peter Schilling ✉ 

Zurich University of Applied Sciences ZHAW,
Winterthur, Switzerland

Joanna Weng ✉ 

Zurich University of Applied Sciences ZHAW,
Winterthur, Switzerland

Ricardo Chavarriaga ✉ 

Zurich University of Applied Sciences ZHAW,
Winterthur, Switzerland

Abstract

The trustworthiness of artificial intelligence systems is crucial for their widespread adoption and for avoiding negative impacts on society and the environment. This paper focuses on implementing a comprehensive certification scheme developed through a collaborative academic-industry project. The scheme provides practical guidelines for assessing and certifying the trustworthiness of AI-based systems. The implementation of the scheme leverages aspects from Machine Learning Operations and the requirements management tool Jira to ensure continuous compliance and efficient lifecycle management. The integration of various high-level frameworks, scientific methods, and metrics supports the systematic evaluation of key aspects of trustworthiness, such as reliability, transparency, safety and security, and human oversight. These methods and metrics were tested and assessed on real-world use cases to dependably verify means of compliance with regulatory requirements and evaluate criteria and detailed objectives for each of these key aspects. Thus, this certification framework bridges the gap between ethical guidelines and practical application, ensuring the safe and effective deployment of AI technologies.

2012 ACM Subject Classification Computing methodologies → Artificial intelligence; Social and professional topics → Computing / technology policy; Information systems → Information systems applications

Keywords and phrases AI Assessment, Certification Scheme, Artificial Intelligence, Trustworthiness of AI systems, AI Standards, AI Safety

Digital Object Identifier 10.4230/OASICS.SAIA.2024.15

Category Academic Track

Related Version *Previous Version:* <https://ieeexplore.ieee.org/document/10675927>

Previous Version: <https://digitalcollection.zhaw.ch/server/api/core/bitstreams/488591b2-c384-44d1-b412-18c9b47bd47d/content>

Previous Version: <https://ieeexplore.ieee.org/document/10675803>

Funding This work was co-financed by Innosuisse (101.650 IP-ICT). The contribution of R.C. was partially funded by the Wellcome Trust [Grant number: 226486/Z/22/Z].



© Carmen Frischknecht-Gruber, Philipp Denzel, Monika Reif, Yann Billeter, Stefan Brunner, Oliver Forster, Frank-Peter Schilling, Joanna Weng, and Ricardo Chavarriaga; licensed under Creative Commons License CC-BY 4.0

Symposium on Scaling AI Assessments (SAIA 2024).

Editors: Rebekka Görges, Elena Haedecke, Maximilian Poretschkin, and Anna Schmitz; Article No. 15; pp. 15:1–15:18
OpenAccess Series in Informatics



OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

15:2 AI Assessment in Practice: Implementing a Certification Scheme ...

Acknowledgements We would like to acknowledge the support and collaboration of CertX AG in the development of the certification scheme discussed in this paper.

1 Introduction

Global efforts are underway to implement frameworks for assessing and regulating artificial intelligence (AI) systems. The most imminent of these efforts is the EU Artificial Intelligence Act [17]. The AI act gradually comes into force starting 1 August 2024, which means organisations and certifiers are in dire need of building their capacity to prove and assess compliance now. However, despite this and other forthcoming regulations around the globe, there remains a significant lack of practical guidelines and methodologies for both achieving and assessing the trustworthiness of AI-based systems (AIS). Although there has been extensive development of ethical guidelines for AI, c.f., Jobin et al. [32], the practical application of these principles remains vague. The lack of specificity in the operationalisation of these guidelines presents a challenge to their effective implementation across various AIS. The introduction and deployment of inadequately understood and unreliable AI technologies can result in significant societal harm. These include the exclusion or discrimination of minorities due to inherent biases [37] and even physical injuries resulting from erroneous decision-making by AIS, such as in human-robot interactions or misdiagnoses in the healthcare sector. Furthermore, such technologies have the potential to exacerbate existing educational disparities, lead to unfair legal outcomes, and increase inequality [42]. There is also a substantial risk of environmental damage, privacy breaches, and cybersecurity vulnerabilities. Certain AI models, especially those based on deep neural networks, are known to be vulnerable to adversarial attacks, where subtle modifications to input data can cause significant errors in system behaviour [24]. It is, therefore, imperative to develop tools that allow for AIS to be thoroughly vetted for responsibility and ethical considerations to mitigate these risks and protect societal well-being.

To address this issue, the authors, in collaboration with a certification company, are developing a certification scheme for AIS. This scheme is intended as a practical guide and provides corresponding tools for developers and regulators to evaluate and certify the trustworthiness of AIS throughout their lifecycle, including requirements, data acquisition, model development, testing, deployment, and operation. It builds upon current standards and guidelines of a number of bodies, including ISO/IEC, IEEE, EASA, as well as other guidance documents [31, 28, 16, 47, 40], in addition to EU legislation. A total of 38 documents were subjected to analysis, and the objectives and the various means of complying with them were derived from these inputs.

This certification scheme effectively bridges the gap between regulatory requirements, technical standards, and the specific scientific and technical methods needed to assess the properties of machine learning (ML) models. Noteworthy, regulatory requirements and technical standards do not provide clear instructions on which methods and metrics can be used to assess the properties for trustworthy AIS. To fill this gap, we evaluated and identified 95 technical methods for assessing the transparency, explainability, reliability, robustness, safety, and security of AI models. By doing so, the certification scheme complements existing approaches in trustworthy AI certification by incorporating cutting-edge research from the AI community on algorithmic techniques for determining and evaluating relevant model properties. As a result, it provides a complete operational framework that links the regulatory requirements to measurable objectives and methods to assess compliance with the EU AI Act and supports regulations in other jurisdictions.

C. Frischknecht-Gruber et al.**15:3**

This paper outlines the implementation and application of the certification scheme, with a particular focus on detailing the tools, workflows, and methodologies used to ensure both comprehensive compliance and practical utility. Furthermore, it describes how these tools and methodologies relate to objectives for means of compliance and demonstrates our approach to assessing the given requirements.

Identifying, tracing, and documenting appropriate objectives, procedures, and technical methods for assessing compliance requires adequate supporting tools. We address these needs by implementing the certification scheme within the requirements and project management platform Jira. This is complemented by an automatised pipeline that implements algorithmic methods for assessing the trustworthiness of AI models. This pipeline is implemented according to best practices in AI engineering and Machine Learning Operations (MLOps) principles.

In the remainder of this paper, we give an overview of the current state of AI standardisation and regulatory efforts, highlighting key initiatives and guidelines. In Section 3, we outline the certification scheme, detailing relevant regulatory requirements, criteria, and the methodology for certification.

Then, we describe the implementation process, including tools and frameworks used to verify compliance, and how these relate to the regulatory requirements (Section 4). Finally, we summarise our findings and offer a discussion on the implications and future developments in AI certification (Section 5).

2 Background

The deployment and scalability of AI assessment frameworks face several key challenges, particularly in balancing practical implementation with theoretical underpinnings. One of the main obstacles lies in the aggregation of risks associated with AI systems, including bias, transparency, security vulnerabilities and ethical considerations [53]. Current frameworks often address individual risks in isolation, but aggregating these risks in a way that provides a holistic assessment is complex [5]. Practical challenges in responsible AI implementation, such as balancing transparency, fairness, and robustness, highlight the need for integrated frameworks that address diverse AI risks holistically [8]. Explainability, in particular, plays a critical role in risk aggregation, as it enables stakeholders to interpret AI decision-making processes, fostering trust and accountability [52]. Many frameworks still lack widely accepted methods for this aggregation, leading to inconsistencies across industries and sectors. A significant need for interdisciplinarity also poses a challenge in scaling AI assessment frameworks. Inputs from law, ethics and computer science must be combined to form a coherent assessment approach. Managing this complexity requires the integration of technical AI safety measures with broader societal values, which is often challenging to operationalise at scale [53]. In terms of approaches, the risk-based approach used in regulations such as the EU AI Act offers a promising method for scaling up. This regulation categorises AI systems according to the level of risk they pose, from low-risk applications such as spam filters to high-risk systems such as healthcare AI. The EU AI law imposes strict regulatory requirements on high-risk systems to ensure safety and accountability. Conversely, AI systems classified as low risk are subject to a more flexible regulatory framework. Although these systems are not subject to the same stringent requirements, they must still comply with transparency and user information obligations. Explainable AI techniques play a crucial role in fulfilling these transparency requirements, as they allow users to understand and interpret how AI systems reach their decisions. This transparency fosters trust and enables

15:4 AI Assessment in Practice: Implementing a Certification Scheme ...

users to make informed choices, even in lower-risk applications where direct oversight may be minimal [20, 52]. This risk-based classification ensures that regulatory oversight is aligned with the potential impact of AI systems, thereby increasing overall regulatory effectiveness while facilitating innovation in lower-risk areas [13].

2.1 Regulation and Standards

Currently, there are significant global efforts to establish regulatory frameworks for AI. The EU has assumed a pioneering position with the AI Act, which is designed to establish a comprehensive regulatory framework for AIS [17]. In a similar vein, the United States issued an executive order in October 2023 with the objective of developing new standards for safe, secure, and trustworthy AI [59].

Standards and guidelines play a pivotal role in supporting binding laws and regulations by documenting best practices and providing a foundation for demonstrating compliance and certification. A considerable number of national and international organisations are engaged in a range of initiatives aimed at fostering trust in AI through the issuance of standards and guidelines. Several ISO/IEC standards [31] are currently being developed to address AI-related aspects, including terminology, performance metrics, data quality, ethics, and human-AI interaction. Some of these standards have already been released, with more anticipated in the future. Similarly, the IEEE is developing a certification program with the objective of assessing the transparency, accountability, bias, and privacy of AI-related processes [26]. The IEEE P7000 series [28] addresses the ethical implications of AI technologies. The European standard development organizations (CEN/CENELEC) have been tasked to develop the standards that will be used to assess conformity with the EU AI Act. As part of this process, they will identify and adopt international standards already developed or under development [10]. Other national entities, such as the National Laboratory of Metrology and Testing's (LNE) AI certification program, have established objective criteria for trustworthy AIS, emphasising ethics, safety, transparency, and privacy [35]. The NIST framework [40] offers guidance on the management of risks, the assurance of data quality, and the promotion of transparency and accountability in AIS, with related principles also emphasised in the AI Risk Management Framework [41]. Moreover, DIN/DKE offers comprehensive standardisation recommendations across all AI domains, facilitating a unified language, principles for development and utilisation, and certification [15]. In the field of aviation, the European Union Aviation Safety Agency (EASA) has introduced comprehensive guidelines for the safe utilisation of ML systems [58]. These guidelines provide support to stakeholders in the aviation sector at each stage of the lifecycle of AIS, from the initial stages of development through to operational use. The Fraunhofer Institute has developed a guideline for the design of trustworthy AI systems [47]. The guideline employs a six-dimensional evaluation framework to assess the trustworthiness of AIS, encompassing fairness, autonomy and control, transparency, reliability, safety and security, and privacy. In contrast to other contributions, the Fraunhofer guideline incorporates both process-related measures and technical methods to enhance the evaluation of AIS.

2.2 Frameworks

Capturing, tracing, documenting, and systematically evaluating requirements throughout the lifecycle of an AIS is an essential factor in trustworthy AI and its certification.

There are various methods and tools for the filtering and management of requirements, from very basic text files or Excel sheets to dedicated frameworks such as Confluence, Jira, Doorstop, Polarion, IBM Doors, Azure DevOps, and many more [4, 3, 7, 56, 48, 39]. In

practice, the simple solutions do not provide the necessary flexibility and overview of the complicated relations between requirements. On the other hand, comprehensive requirement management frameworks are flexible but often less intuitive in their use and relatively expensive. After investigating several tools, we chose **Jira** (in its basic version, free) as a requirement management tool [3] for the certification of AIS. Jira is a project management and issue-tracking software developed by Atlassian. It helps teams plan, track, and manage work efficiently, offering features like customisable workflows, real-time reporting, and integration with numerous other tools, making it a versatile solution for agile project management.

An important operational approach to scaling is the integration of Machine Learning Operations (MLOps). The role of MLOps principles and best practices in AIS development and operation, as well as its assessment, is twofold: First, Billeter et al. [6] and others [36] have advanced the idea of MLOps as an enabler of trustworthy AI by design. This means that following MLOps guidelines and principles during design, development and operation of an AIS, will lead to increased trustworthiness of the AIS. These practices include version control, continuous integration and deployment (CI/CD), automated testing, and monitoring. Second, the assessment of the trustworthiness of AIS also requires comprehensive evaluations of many objectives and means of compliance (MOC) derived from these requirements. Therefore, concepts like following best practices in AI engineering and MLOps are indispensable not just during AIS development but also during its assessment.

2.3 Algorithmic Tools for Trustworthy AI

While in some aspects of the verification of AI trustworthiness it is necessary to rely on qualitative results, in particular for model explainability or robustness, automated evaluation workflows mostly involve algorithmic methods with quantifiable output. Therefore, it is crucial to integrate assessment toolboxes which implement various algorithms and metrics, or rely on interfaces which allow for manual qualitative evaluation. There are a number of comprehensive toolboxes which implement appropriate technical methods paired with metrics, often isolated to assess specific aspects of AI trustworthiness such as transparency, reliability, or safety. For data and model explainability, industry-developed frameworks include Microsoft's InterpretML [38], Seldon's Alibi toolbox [54], IBM's AIX360 toolbox [61], Sicara's tf-explain [55], or PyTorch's captum API [9]. Additionally, Quantus [25] is a relatively new and complementary explainability toolbox which implements a growing number of metrics and provides interfaces for other toolboxes such as captum or tf-explain. Toolboxes for testing the reliability, robustness, and safety of an AI model are, e.g., MIT's Responsible AI Toolbox [57], Seldon's Alibi-Detect [63], IBM's ART [60] and UQ360 [62] toolboxes. In particular, there are many more toolboxes which implement specific tests for adversarial robustness, such as RobustnessGym [23], CleverHans [46], or Foolbox [49].

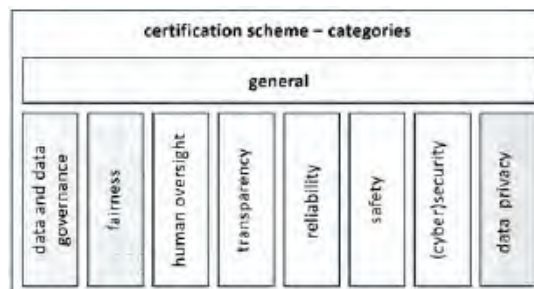
It is worth noticing that these toolboxes have been developed in parallel and, to a large extent, disconnected from the regulatory and certification frameworks. Hence, their suitability for compliance assessment is not entirely clear. Our analysis presents a significant step towards the integration of advances on both areas.

3 Overview of the Certification Scheme

The developed certification scheme for AIS encompasses several principal key aspects of trustworthiness, such as human oversight, transparency, and robustness, which we have defined more granularly by including safety, security, and reliability [14], as illustrated in Figure 1. While we considered the trustworthiness dimensions from the EU, as described in

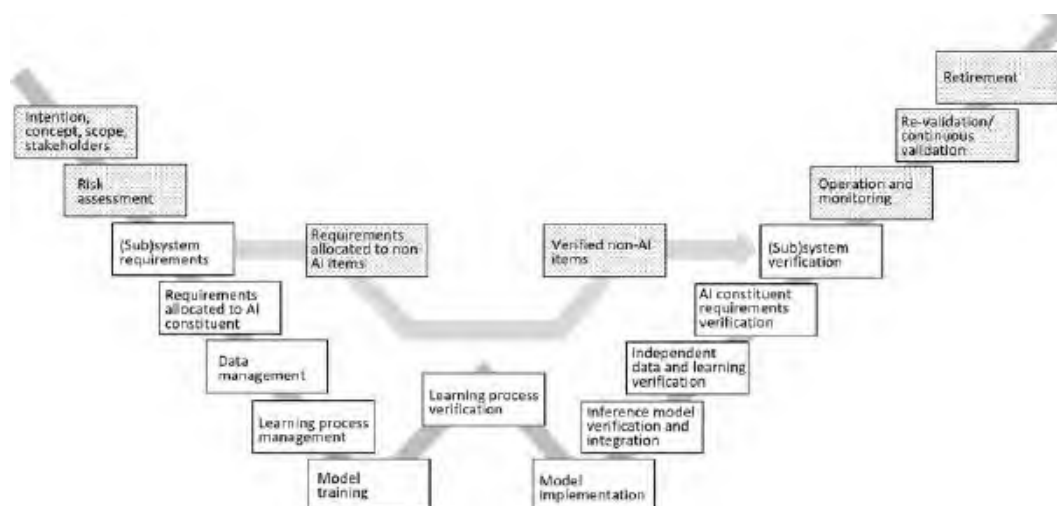
15:6 AI Assessment in Practice: Implementing a Certification Scheme ...

ALTAI [43], we opted for this more detailed breakdown to enable more precise tracking of objectives. Each aspect is considered to ensure that AIS operate effectively, ethically, and safely across various applications.



■ **Figure 1** Extended key aspects of trustworthiness. The aspects of trustworthiness are as follows: data and data governance, fairness, human oversight, transparency, reliability, safety, (cyber)security, and data privacy. Within the current version of certification scheme, the five non-shaded aspects are addressed, while the other three will be addressed at a later stage.

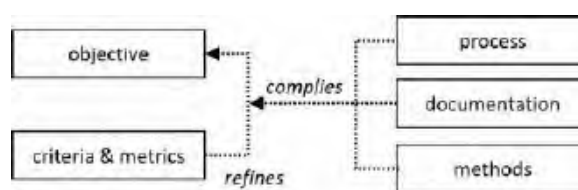
In addition, the certification scheme encompasses all relevant phases of the AIS lifecycle, as illustrated in Figure 2. The Certification Scheme employs a risk-based methodology in accordance with the EU AI Act by assessing each AIS as a minimal, limited, or high-risk level application and tailoring the evaluation rigour accordingly. The applicant seeking certification performs and provides the initial risk assessment for their AIS, which then the certifier verifies to ensure alignment with the definitions and criteria set forth in the Act. In addition, risk assessments are based on standards or best practices within different industries and address any additional risks if applicable. The certification process commences with the concept of the system (including the role of the AI part within the overall system), including understanding its role within the larger operational context and determining its associated system risk, subsequently progressing to the implementation of an AI model. The scheme culminates with the deployment, verification, validation, and operation of the system.



■ **Figure 2** Illustration of the lifecycles encompassed by the certification scheme, including risk assessment, (sub-)system requirements and design, data management, learning process management, model training, verification steps, and operation and monitoring.

3.1 Key Aspects and Objectives

For each phase of the lifecycle, the scheme identifies and addresses the critical key aspects through the establishment of corresponding objectives. In collaboration with the certification body CertX [11], we conducted a targeted analysis of 38 key documents to establish a robust foundation for the certification. These documents, selected in light of the fact that many regulations are still forthcoming and that numerous standards remain under development and are not yet active, were chosen based on their relevance to existing regulations, technical standards, and guidance materials essential for ensuring trustworthiness of AIS. The selection encompassed recognised standard bodies and authoritative guidelines, such as those from the ISO/IEC Joint Technical Committee on Artificial Intelligence (ISO/IEC JTC 1/SC 42) [31], the IEEE Autonomous and Intelligent Systems (AIS) Standards [27] and EASA [58], as well as EU legislative requirements and the Artificial Intelligence Standardization Roadmap developed by DIN and DKE [15], among other sources. The objectives are refined according to different qualitative criteria and quantitative metrics (see Figure 3). Different MOCs have been defined to achieve compliance with the aforementioned objectives. One group of MOCs describes the process means that must be in place for a thorough development, verification, or management process. Others describe the documentation means to cover, for example, auditability and other record-keeping aspects. The last group of MOCs define the technical methods that must be applied to achieve compliance with the objectives posed. These MOCs establish the link to the different technical methods of the second technical part of the certification scheme.



■ **Figure 3** The interrelationship between objectives, criteria and metrics, and compliance methods is illustrated in the diagram. The left side depicts the objectives and their refinement through the application of criteria and metrics, while the right side shows the processes, documentation and methods that ensure compliance with these objectives and criteria.

Initially, the certification scheme focused on transparency and reliability, encompassing 29 and 44 objectives, respectively, with 100 and 156 MOCs. An updated version of the scheme additionally includes human oversight, safety and security alongside some general objectives relevant across multiple key aspects. Currently, the scheme covers:

- General Objectives: 5 objectives, 14 MOCs
- Human Oversight: 62 objectives, 65 MOCs
- Transparency: 29 objectives, 53 MOCs
- Reliability: 36 objectives, 105 MOCs
- Safety: 2 objectives, 6 MOCs
- (Cyber)Security: 5 objectives, 17 MOCs

The scheme includes a risk analysis and also addresses overlapping areas across key aspects, ensuring a comprehensive and integrated approach. Additional key aspects, such as data and data governance, will be implemented in the next step, and the key aspects of fairness and data privacy are planned for subsequent steps. In the following, we present two example objectives and their corresponding MOCs.

15:8 AI Assessment in Practice: Implementing a Certification Scheme ...

Objective 1. The applicant should define performance metrics to evaluate AIS performance and reliability.

- **MOC:** Define a suitable set of performance metrics for each high-level task to evaluate AIS performance and reliability.
- **MOC:** Define the expected performance with training, validation, and test data sets.
- **MOC:** Provide a comprehensive justification for the selection of metrics.

Objective 2. The applicant should identify and document the methods at AI/ML item and/or output level satisfying the specified AI explainability needs.

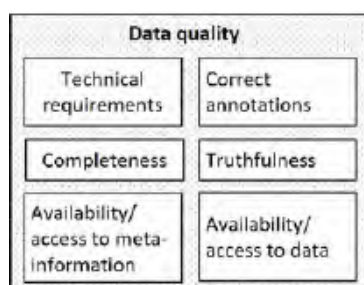
- **MOC:** Provide documentation of methods to provide explanations about the AI/ML item. The type and scope of the provided explanations should be chosen in terms of proportionality, considering the stakeholders.
- **MOC:** Specify the rules that apply to the current decision (e.g., for decision trees, list the selected branching next to the model output).
- **MOC:** Specify the most relevant attributes for a decision in linear regression models (e.g., for normalised inputs, the largest absolute coefficient value).
- **MOC:** For white-box models, use model-specific or model-agnostic methods for interpretability.

3.2 Key Aspects Overview

This section provides an overview of the key aspects covered by the scheme, including data governance, human oversight, transparency, reliability, and safety and (cyber)security.

3.2.1 Data and Data Governance

A dependable data set for a specific task requires careful attention to four key aspects: data quality, completeness, representativeness, and transparency [29, 30, 1, 19]. Data quality focuses on ensuring formal completeness and correctness and establishing reliability. The training, validation, and test data quality is assessed through qualitative and quantitative means (Figure 4). Correct annotations, task relevance, and data origin are crucial, alongside ensuring application coverage through metrics like class balance. Bias prevention requires unbiased training, validation, and test data, with fairness assessed via metrics like cosine similarity. Guidelines like the NIST AI Risk Management Framework outline methods to minimise bias and ensure fairness [41]. Transparency ensures data is interpretable and preprocessing steps are clear, enabling verification by stakeholders.



(a) Data quality consists of six aspects.



(b) Data coverage for the application.

■ **Figure 4** Data quality (formal data completeness and correctness) and data coverage.

3.2.2 Human Oversight

Human oversight of AIS, also referred to as autonomy and control, addresses potential risks that may arise when autonomous AI components limit the ability of users or experts to perceive or act. This aspect of AI safety ensures that system autonomy is appropriately constrained when it deviates from normal operation. To assess human oversight, AIS are categorised into four levels based on human involvement [44]. The first level, Human Control (HC), involves the AI acting solely as an assistive tool, where humans are responsible for every decision and subsequent action based on the AI's output. At the Human-in-the-Loop (HIL) level, the AI operates partially autonomously but requires human intervention or confirmation, with humans monitoring and correcting its decisions as needed. The Human-on-the-Loop (HOL) level allows the AI to function almost autonomously, with limited human involvement for monitoring and occasional overrides. Finally, at the Human-out-of-the-Loop (HOOTL) level, the AI operates fully autonomously, handling tasks independently even in unexpected situations, with humans only involved in initial setup decisions like setting meta-commands in autonomous vehicles.

This key aspect includes objectives such as the implementation of human monitoring and control mechanisms, preservation of human decision-making capabilities, and ensuring the traceability of the AI component's decision-making process.

3.2.3 Transparency

Transparency in AI is essential to prevent potential harm and ensure systems are understandable to different stakeholders [51]. Transparency objectives are tailored to users, those affected (society), and experts (developers, providers, auditors and evaluators, authorities) (Figure 5). It involves setting criteria for interpretability and explainability, focusing on clarity, comprehensibility, and relevant metrics [12]. The interpretability of the ML model must be ensured through thorough documentation and visual aids like schematic diagrams. Explanation methods should be carefully chosen, justified, and documented, considering the audience's qualifications. These methods should be evaluated statistically and by human reviewers, with a system in place for addressing user queries. For experts, transparency also involves validating decisions, ensuring technical traceability, and maintaining reproducibility. Key considerations include the scope, design, and stability of explanation methods relative to model outputs.

Society: Trust and understanding by clearly communicating the strengths and limitations.	Developers: Clear insights into the internal workings, and limitations.	Users: Transparent explanation of the decisions and results.	Authority: Assurance of regulatory compliance and operational transparency by thorough documentation.
	Providers: Monitoring capability by information on internal operations and performance	Auditors & Evaluators: Audit/evaluation capability	

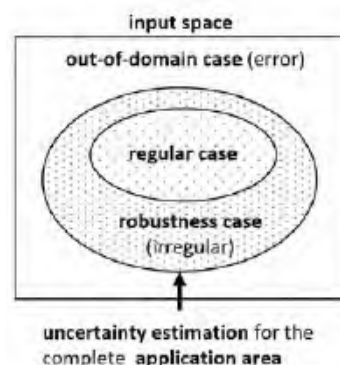
■ **Figure 5** Transparency needs vary between stakeholders. The figure shows exemplary transparency requirements for some key stakeholders.

3.2.4 Reliability

Reliability in AIS is defined as the consistent execution of intended functions and also entails robustness, which pertains to maintaining performance under disturbances. An important concept is the Operational Design Domain (ODD), which delineates the specific conditions

15:10 AI Assessment in Practice: Implementing a Certification Scheme ...

under which AIS can operate safely and effectively [50]. For developers, the ODD builds the basis for deriving detailed technical specifications that define the AIS input space, categorised into regular cases involving minor, expected disturbances; robustness cases where larger disturbances are encountered; and out-of-domain (OOD) cases, which involve data outside the application domain which may result in errors (Figure 6).



■ **Figure 6** Visualisation of the input space divided into regular, robustness, and out-of-domain cases.

Consequently, reliability is assessed in the three input spaces, in addition to the estimation of uncertainty. The regular case ensures reliable performance through data coverage, augmentation, and performance metrics evaluation (see Figure 7). Robustness tackles challenging conditions by addressing vulnerabilities and adversarial attacks [30, 19]. In out-of-domain (OOD) cases, the focus is on catching errors and improving generalisation, while uncertainty estimation involves setting appropriate metrics, assessing both intrinsic and extrinsic uncertainties, and developing mitigation measures.

Performance metrics			
Regression	likelihood ratio	Confusion matrix	Completeness score
• (Mean) squared error	• True/False-negative rate	• Hinge loss	
• (Mean) absolute error	• True/False-positive rate	Computer vision	Ranking
Classification	• Precision-recall curve	• Peak-signal-to-noise ratio (PSNR)	• Mean reciprocal rank
• (Balanced) accuracy	• Receiver operation characteristics (ROC)	• Structural similarity	• Discounted cumulative gain
• (Balanced) F1-score	• Lift	• (Mean) intersection over union (mIOU)	Natural Language Processing
• Prevalence	• Matthew's correlation coefficient	Clustering	• Perplexity score
• Precision	• Area under curve (AUC)	• Silhouette value	• BLEU score
• False discovery/omission rate	• Cohen's Kappa	• Adjusted mutual information score	
• Positive/negative			

■ **Figure 7** Non-exhaustive list of performance metrics used for regression, classification, computer vision, clustering, ranking, and natural language processing.

Additional process steps include evaluating model architecture, implementing optimisation techniques such as pruning or quantisation, ensuring reproducibility, conducting regular assessments, and meticulously documenting all activities.

In the certification scheme, reliability assessment involves over 55 metrics and 95 methods, with a subset of 35 metrics and 50 methods selected for empirical testing. This selection was based on relevance, execution time, reliance on available information, and computational costs.

Metrics vary across application domains and model objectives, so choosing the appropriate metric and method requires careful consideration of the model's goals, data characteristics, and desired outcomes. For example, formal verification employs logical and mathematical proofs to confirm system criteria, while model coverage analysis ensures comprehensive testing across various scenarios.

3.2.5 Safety and (Cyber)Security

The objective of safety is to minimise harm to people and the environment by designing AIS that incorporate corrective mechanisms for unexpected behaviours [30]. This is of particular importance in contexts such as autonomous vehicles and healthcare, where errors can have significant and adverse consequences. (Cyber)security guarantees a system's integrity and availability by safeguarding it against unauthorised access, modification, or destruction [18]. This encompasses the implementation of robust access controls, the assurance of data and model integrity, and the maintenance of system availability even in the event of an attack. Effective security measures are imperative for AIS in critical infrastructure, as breaches could result in significant damage. In order to enhance the security and resilience of AIS, adversarial training and verification are employed. Adversarial training is a method for enhancing the robustness of a model by exposing it to perturbations designed to deceive it, thereby identifying potential vulnerabilities [24].

4 Implementation of the Certification Scheme

The implementation and subsequent application of the AIS Certification Scheme to customers must meet established standards and regulations, requiring a carefully managed process, including adherence to the EU AI Act, standards from ISO/IEC and IEEE [30, 17, 26, 44], among others. To achieve this, we evaluated several requirements management tools and ultimately selected Jira as the tool for organising, documenting, and tracing the objectives and associated means of compliance for our certification scheme. We then implemented an MLOps system based on state-of-the-art open-source tooling to perform the technical assessment of the AIS and evaluate compliance with the defined objectives. In the following, we describe the requirement management system and the MLOps infrastructure.

4.1 Requirement Management Implementation

As written in section 2.2, Jira was chosen as requirement management tool to ensure traceability and effective management of the requirements. AI certification frameworks must adhere to internationally recognised standards, including ISO 9001 (Quality Management Systems), ISO/IEC 27001 (Information Security Management Systems), and the ISO/IEC 23894 (AI - Guidance on Risk Management). Such standards necessitate meticulous documentation, traceability, and periodic auditing to guarantee sustained compliance. The implementation of such requirements in a manual or disparate system would increase the risk of inconsistencies and errors, which would ultimately impact the efficiency and credibility of the certification process. It is therefore imperative that robust requirements management tools are employed.

Using Jira for requirement management ensures each objective and MOC is meticulously organised, facilitating clear communication and comprehensive oversight. Its ability to maintain detailed records and provide real-time updates is crucial for this task. Real-time collaboration and review capabilities are critical in aligning project tasks and reducing errors. The platform supports multi-user editing, allowing teams to work simultaneously

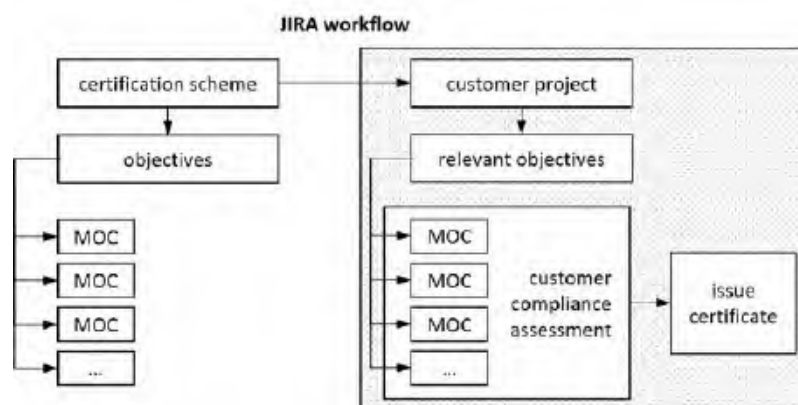
15:12 AI Assessment in Practice: Implementing a Certification Scheme ...

from different locations. This live collaboration and features, such as decision tracking and impact analysis, ensure that the development of the certification scheme remains agile and responsive to changes. Additionally, the system's version control and history management provide a complete audit trail, which is crucial for maintaining consistency and verifiability.

Centralised management of objectives and MOCs in a digital environment allows for streamlined workflows and task alignment. We developed customised templates and dashboards for managing and tracing requirements. The possibility of sorting issues by attributes such as the tag COMPLETE was proven to facilitate requirement tracking in the evaluation we made of the platform. Each objective and MOC can be linked to others, showing relationships such as blocking issues and dependencies. The system's adaptability through the reusability of issues across different projects and its capacity for baseline creation significantly enhance the efficiency of the certification process. The platform facilitates organised and efficient project management by enabling tasks such as editing, organising decision-making, and managing tasks through a user-friendly interface. Integration with state-of-the-art tools, such as Git integration platforms like GitHub or GitLab, as well as business communication tools like Teams and Slack, along with the capability to create customisable pages, allows the tool to be precisely tailored to specific project needs.

The certification scheme is structured with a parent-child relationship between objectives and MOCs (Figure 8). Each issue type is defined by attributes, including description, main category, additional categories, lifecycle phase, risk level, references, and approval status, ensuring thorough documentation and easy information retrieval via specific filtering. This structured approach facilitates organisational efficiency and enables the certification process to be adapted as required.

For practical use, the certification scheme we developed has been implemented as a base project; which can be readily exported, adapted, re-imported, or cloned to align with the particular requirements of the customer or AI system to be assessed. For certification bodies working with clients, the base project serves as the foundation from which the customer's certification project is derived. The customer's AIS is then assessed against the MOCs from the base scheme, supporting the issuance of the final certification.



■ **Figure 8** Visualisation of the certification workflow based on JIRA.

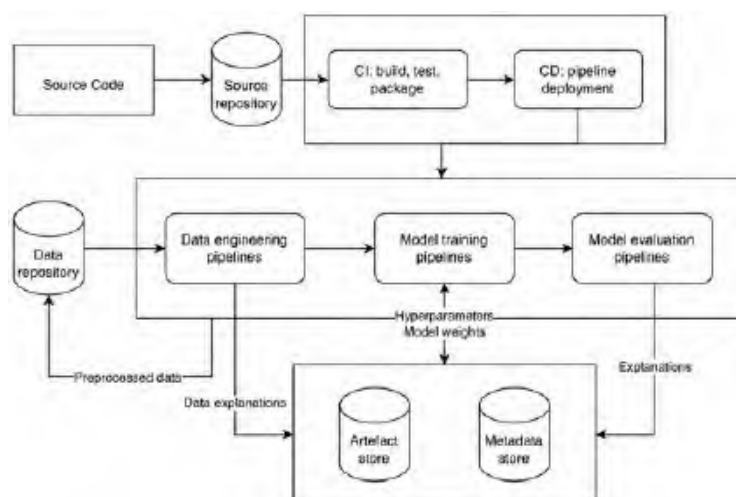
4.2 Machine Learning Operations Infrastructure

As argued in section 2.2, MLOps serves as an enabler for trustworthy AI by design. It provides the necessary infrastructure and practices, ensuring that AIS are developed, deployed, and maintained reliably and efficiently. The adoption of MLOps thus facilitates the integration of

trustworthy AI principles at every stage of the AI lifecycle, which are critical for regulatory compliance and societal acceptance [6]. MLOps extends DevOps practices to manage the complexities of bringing AIS into production, ensuring that they continuously meet trustworthiness standards [33].

The general architecture an AIS which adheres to MLOps principles supports the entire lifecycle of AIS and ensures that models are reproducible, reliable, and maintainable. This architecture includes project setup and requirements engineering, data engineering, model development, continuous integration/continuous deployment (CI/CD), and monitoring and maintenance. The requirement management system described in Section 4.1 will be part of the project setup and requirements engineering phase.

Complementing the requirements management, we implemented a full software pipeline for implementation, training and validation of AI models. This pipeline could be used (a) by AI developers for continuously tracking compliance with the certification requirements, or (b) by certifiers to perform systematic tests of their clients AI models. It thus demonstrates the benefits of MLOps best practices in terms of trustworthiness by design, implemented in the AIS development, as well as in terms of facilitating an efficient means of compliance tracking and verification as part of a certification.



■ **Figure 9** Overview of the MLOps system architecture.

In our pipeline, models are developed, trained and versioned using Git (through GitHub [21]) and MLflow [34], which document all changes to the models' code and parameters, respectively. MLflow also provides the tooling for tracking experiments, packaging code, and managing model deployment. The model development process involves experimentation with different algorithms and hyperparameters to optimise performance. GitHub Actions [22] and Apache Airflow [2] are used for workflow scheduling and monitoring and facilitate automated testing and deployment processes in CI/CD pipelines. Data is versioned using Oxen [45]. A schematic of the system is shown in Figure 9. The system listens for modifications to the model source code or input dataset. Changes automatically trigger training and model evaluation pipelines, which execute tests based on the methods described in sections 3.2.3, 3.2.4, and 3.2.5. For the certification scheme, we mainly relied on methods from captum, Alibi, AIX360, ART, and UQ360, as well as original implementations from academic papers. The outputs, model parameters and similar artefacts, are stored and versioned. Additionally, data engineering pipelines are run, which prepare the data for training and evaluation, and perform data-related trustworthiness evaluations.

15:14 AI Assessment in Practice: Implementing a Certification Scheme ...

MLOps provides several benefits to both AIS development and certification. Traceability and documentation are maintained throughout the AI lifecycle, providing a clear audit trail and ensuring that all objectives and means of compliance are systematically recorded. Version control is critical to maintaining the integrity of AI models and datasets, allowing teams to revert to previous versions if necessary and ensuring that all changes are documented and traceable. Automation and testing are streamlined through CI/CD pipelines, ensuring that each change is rigorously tested for compliance with trustworthiness standards before deployment. Post-deployment, continuous monitoring of AIS ensures that they remain compliant and perform reliably in real-world conditions.

Our workflow and methods have been tested in two real-world computer vision use cases in medical applications and vehicle detection on construction sites [14]. These use cases correspond to distinct high-risk applications according to the EU AI act. These use cases provide a test bed for validating the tools for certification on different data types and sets of requirements.

5 Discussion

The proposed certification scheme introduces several significant innovations in the assessment and certification of AIS trustworthiness, addressing an important gap in current practices. Despite the existence of standards, ethical guidelines and regulations, there remains a significant gap in the availability of practical tools and methodologies to achieve and systematically assess compliance. Our certification scheme addresses this gap by providing structured tools that are crucial for the rigorous evaluation of AIS. The scheme is underpinned by an extensive review and integration of 38 key documents from various standards and regulatory bodies, such as ISO/IEC, IEEE, EASA, and the Fraunhofer Institute. This foundational research ensures that the certification objectives and their means of compliance are comprehensive and aligned with the best practices and requirements across industries.

An important aspect of the scheme's implementation was evaluating multiple requirements management tools to support the certification workflow. Jira was selected for its robust capabilities in managing the complex certification process. This choice was crucial for maintaining systematic tracking of compliance objectives, ensuring that every requirement is meticulously documented and traceable.

Moreover, the means of compliance entail the application of metrics and technical methods by the customer, which can also be employed in the technical assessment of the AIS. Consequently, the scheme incorporates a technical assessment based on the implementation of selected technical methods which are linked to the objectives. The selection is determined through an evaluation of 95 well-established and cutting-edge methods, with the evaluation criteria being their suitability in meeting the defined objectives, criteria, and metrics. These methods were rigorously selected and empirically tested to ensure they provide effective compliance across various key aspects of trustworthiness, such as human oversight, transparency, safety, and (cyber)security. The workflow and methods developed within the certification scheme were tested on two real-life use cases: skin lesion classification and vehicle detection on construction sites. These practical applications demonstrate the scheme's effectiveness and adaptability in diverse, real-world scenarios. In addition, an automated workflow was implemented on a computing cluster following MLOps principles and best practices. This workflow maps MLOps stages with Trustworthy AI principles and key aspects, ensuring continuous compliance and efficient lifecycle management. By automating the certification process, the scheme enhances reliability, reduces human error, and ensures that the certification remains

up-to-date with the latest developments in AI and ML technologies. Also, due to the dynamic nature of AIS and their complex post-deployment environments, trust levels can fluctuate. Continuous risk monitoring is essential to maintain trustworthiness, which is in line with the iterative nature of MLOps and is driven by versioning, automation, testing, deployment, and monitoring. Incorporating trustworthiness metrics alongside traditional performance metrics enables continuous feedback loops that systematically address trustworthiness requirements throughout the AI lifecycle [64].

The primary focus at the beginning of the development of the certification scheme was on reliability and transparency, areas where technical implementations could be more straightforwardly automated. As the scheme has developed, the scope has been expanded to encompass additional key areas, such as human oversight, which present more intricate challenges. These aspects are inherently linked to human interaction, which makes them challenging to automate effectively. The absence of established technical methods and metrics in these areas presents a significant challenge. As an illustration, the assessment of fairness in AI systems is an evolving field with no universally accepted metrics. This makes the certification process more challenging. The scheme provides a structured approach to compliance, whether through design or iterative testing and improvement. However, the absence of reliable metrics makes the implementation process less clear.

The tools and frameworks employed in the implementation of the certification scheme are designed to be adaptable, allowing the scheme to evolve in response to advances in AI techniques and changing requirements. One clear example is the increasing adoption of foundational models (referred to as general-purpose AI models in the EU legislation), including large language models (LLMs). These models, which are trained on vast and diverse datasets, introduce significant complexity due to their context-dependent and sometimes unpredictable behaviour. The subjective nature of their outputs and the difficulty of quantifying their decision-making processes pose challenges for evaluating and validating their trustworthiness within a standardised framework. As these models are increasingly deployed across many use cases, the development of new requirements, MOCs, and methods tailored to these models will be vital. Addressing these challenges will be essential for maintaining the relevance and applicability of the certification scheme as AI technologies continue to advance rapidly.

References

- 1 IEEE 7001-2021 - IEEE Standard for Transparency of Autonomous Systems. Technical report, Institute of Electrical and Electronics Engineers, 2021. URL: <https://standards.ieee.org/standard/7001-2021.html>.
- 2 Apache Software Foundation. Airflow. URL: <https://airflow.apache.org/>.
- 3 Atlassian. Jira, 2002. URL: <https://www.atlassian.com/software/jira>.
- 4 Atlassian. Confluence, 2004. URL: <https://www.atlassian.com/software/confluence>.
- 5 Richard Benjamins, Alberto Barbado, and Daniel Sierra. Responsible AI by design in practice. *arXiv preprint arXiv:1909.12838*, 2019.
- 6 Yann Billeter, Philipp Denzel, Ricardo Chavarriaga, Oliver Forster, Frank-Peter Schilling, Stefan Brunner, Carmen Frischknecht-Gruber, Monika Ulrike Reif, and Joanna Weng. MLOps as enabler of trustworthy AI. In *11th IEEE Swiss Conference on Data Science (SDS), Zurich, Switzerland, 30-31 May 2024*, 2024. doi:10.21256/zhaw-30443.
- 7 Jace Browning and Robert Adams. Doorstop: Text-based requirements management using version control, 2014. doi:10.4236/jsea.2014.73020.
- 8 Miles Brundage, Shahar Avin, Jasmine Wang, Haydn Belfield, Gretchen Krueger, Gillian Hadfield, Heidy Khlaaf, Jingying Yang, Helen Toner, Ruth Fong, et al. Toward trustworthy AI development: mechanisms for supporting verifiable claims. *arXiv preprint arXiv:2004.07213*, 2020.

15:16 AI Assessment in Practice: Implementing a Certification Scheme ...

- 9 Captum. Model interpretability for PyTorch, 2023. URL: <https://captum.ai/>.
- 10 CEN-CENELEC. Artificial Intelligence. URL: <https://www.cenelec.eu/areas-of-work/cen-cenelec-topics/artificial-intelligence/>.
- 11 CertX. CertX: First Swiss Functional Safety and Cyber Security Certification Body. URL: <https://certx.com/>.
- 12 Chun Sik Chan, Huanqi Kong, and Guanqing Liang. A comparative study of faithfulness metrics for model interpretability methods. *arXiv preprint arXiv:2204.05514*, 2022. doi:10.48550/arXiv.2204.05514.
- 13 Council of European Union. Artificial Intelligence Act: Council and Parliament Strike a Deal on the First Rules for AI in the World, 2023. URL: <https://www.consilium.europa.eu/en/press/press-releases/2023/12/09/artificial-intelligence-act-council-and-parliament-strike-a-deal-on-the-first-worldwide-rules-for-ai/>.
- 14 Philipp Denzel, Stefan Brunner, Yann Billeter, Oliver Forster, Carmen Frischknecht-Gruber, Monika Ulrike Reif, Frank-Peter Schilling, Joanna Weng, Ricardo Chavarriaga, Amin Amini, et al. Towards the certification of AI-based systems. In *11th IEEE Swiss Conference on Data Science (SDS), Zurich, Switzerland, 30-31 May 2024*, 2024. doi:10.21256/zhaw-30439.
- 15 DIN, DKE. Artificial intelligence standardization roadmap, 2023. URL: <https://www.dke.de/en/areas-of-work/core-safety/standardization-roadmap-ai>.
- 16 EASA and Daedalean. Concepts of Design Assurance for Neural Networks (CoDANN) II. Technical report, May 2021. URL: <https://www.easa.europa.eu/en/downloads/128161/en>.
- 17 European Parliament and Council of the European Union. Regulation (eu) 2024/1689 of the european parliament and of the council of 9 october 2024 on harmonized rules for artificial intelligence (artificial intelligence act) and amending certain union legislative acts, 2024. URL: <https://eur-lex.europa.eu/eli/reg/2024/1689/oj>.
- 18 International Organization for Standardization. ISO/IEC 27001:2013 information technology – security techniques – information security management systems – requirements. Technical report, ISO, 2013.
- 19 International Organization for Standardization. ISO/IEC 24029-1:2021 Artificial intelligence (AI) — Assessment of the robustness of neural networks — Part 1: Overview. Technical report, International Organization for Standardization, 2021. URL: <https://www.iso.org/standard/77609.html>.
- 20 Leilani H Gilpin, David Bau, Ben Z Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. Explaining explanations: An overview of interpretability of machine learning. In *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*, pages 80–89. IEEE, 2018. doi:10.1109/DSAA.2018.00018.
- 21 GitHub, Inc. GitHub . URL: <https://github.com/>.
- 22 GitHub, Inc. GitHub Actions. URL: <https://github.com/features/actions>.
- 23 Karan Goel, Nazneen Fatema Rajani, Jesse Vig, Zachary Taschdjian, Mohit Bansal, and Christopher Ré. Robustness gym: Unifying the NLP evaluation landscape. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstrations*, pages 42–55, Online, June 2021. Association for Computational Linguistics. URL: <https://www.aclweb.org/anthology/2021.naacl-demos.6>, doi:10.18653/V1/2021.NAACL-DEMOS.6.
- 24 Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples. 2014. doi:10.48550/arXiv.1412.6572.
- 25 Anna Hedström, Leander Weber, Daniel Krakowczyk, Dilyara Bareeva, Franz Motzkus, Wojciech Samek, Sebastian Lapuschkin, and Marina Marina M.-C. Höhne. Quantus: An explainable AI toolkit for responsible evaluation of neural network explanations and beyond. *Journal of Machine Learning Research*, 24(34):1–11, 2023. URL: <http://jmlr.org/papers/v24/22-0142.html>.
- 26 IEEE. IEEE CertifAIED: the mark of AI ethics, 2022. URL: <https://engagestandards.ieee.org/ieeecertifaiied.html>.

C. Frischknecht-Gruber et al.

15:17

- 27 IEEE Standards Association. IEEE Autonomous and Intelligent Systems Standards. URL: <https://standards.ieee.org/initiatives/autonomous-intelligence-systems/standards/>.
- 28 IEEE Standards Association. IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems, 2023. URL: <https://standards.ieee.org/industry-connections/ec/autonomous-systems/>.
- 29 International Organization for Standardization. ISO/IEC 25024:2015 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Measurement of data quality. Technical report, 2015. URL: <https://www.iso.org/standard/35746.html>.
- 30 International Organization for Standardization. ISO/IEC 24028:2020 Information technology — Artificial intelligence (AI) — Overview of trustworthiness in AI. Technical report, 2020. URL: <https://www.iso.org/standard/77608.html>.
- 31 ISO. ISO/IEC JTC 1/SC 42 Artificial Intelligence, 2023. URL: <https://www.iso.org/committee/6794475.html>.
- 32 Anna Jobin, Marcello Ienca, and Effy Vayena. The Global Landscape of AI Ethics Guidelines. *Nature Machine Intelligence*, 1(9):389–399, 2019. doi:10.1038/s42256-019-0088-2.
- 33 Dominik Kreuzberger, Niklas Kühl, and Sebastian Hirschl. Machine Learning Operations (MLOps): Overview, Definition, and Architecture. *IEEE Access*, 11:31866–31879, 2023. doi:10.1109/ACCESS.2023.3262138.
- 34 LF Projects, LLC. MLFlow. URL: <https://mlflow.org/>.
- 35 LNE. Certification of processes for AI, 2023. URL: <https://www.lne.fr/en/service/certification/certification-processes-ai>.
- 36 Beatriz M. A. Matsui and Denise H. Goya. Mlops: A guide to its adoption in the context of responsible ai. In *2022 IEEE/ACM 1st International Workshop on Software Engineering for Responsible Artificial Intelligence (SE4RAI)*, pages 45–49, 2022. doi:10.1145/3526073.3527591.
- 37 Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM computing surveys (CSUR)*, 54(6):1–35, 2021. doi:10.1145/3457607.
- 38 Microsoft. InterpretML. URL: <https://github.com/interpretml/interpret>.
- 39 Microsoft. Azure devops, 2005. URL: <https://azure.microsoft.com/en-us/products/devops/#overview>.
- 40 NIST. NIST Technical AI Standards, 2023. URL: <https://www.nist.gov/artificial-intelligence/technical-ai-standards>.
- 41 NIST. AI Risk Management Framework (AI RMF) Knowledge Base, 2024. URL: https://airc.nist.gov/AI_RMF_Knowledge_Base/AI_RMF.
- 42 Office of the United Nations High Commissioner for Human Rights (OHCHR). A taxonomy of AI and human rights harms. Technical report, United Nations Human Rights Office of the High Commissioner, 2023. Accessed: 2024-11-06. URL: <https://www.ohchr.org/sites/default/files/documents/issues/business/b-tech/taxonomy-GenAI-Human-Rights-Harms.pdf>.
- 43 High-Level Expert Group on Artificial Intelligence. Assessment list for trustworthy artificial intelligence (ALTAI), 2020. URL: <https://altai.insight-centre.org>.
- 44 Independent High-Level Expert Group on Artificial Intelligence. Ethics guidelines for trustworthy AI. Technical report, European Commission, 2019. URL: <https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai>.
- 45 Oxen.ai. oxen. URL: <https://www.oxen.ai/>.
- 46 Nicolas Papernot, Fartash Faghri, Nicholas Carlini, Ian Goodfellow, Reuben Feinman, Alexey Kurakin, Cihang Xie, Yash Sharma, Tom Brown, Aurko Roy, Alexander Matyasko, Vahid Behzadan, Karen Hambardzumyan, Zhishuai Zhang, Yi-Lin Juang, Zhi Li, Ryan Sheatsley, Abhibhav Garg, Jonathan Uesato, Willi Gierke, Yinpeng Dong, David Berthelot, Paul Hendricks, Jonas Rauber, and Rujun Long. Technical report on the cleverhans v2.1.0 adversarial examples library. *arXiv preprint arXiv:1610.00768*, 2018.

15:18 AI Assessment in Practice: Implementing a Certification Scheme ...

- 47 Maximilian Poretschkin et al. Leitfaden zur Gestaltung vertrauenswürdiger Künstlicher Intelligenz (KI-Prüfkatalog), 2021. URL: <https://www.iais.fraunhofer.de/de/forschung/kuenstliche-intelligenz/ki-pruefkatalog.html>.
- 48 Rational Software. IBM doors, 2018. URL: <https://www.ibm.com/docs/en/engineering-lifecycle-management-suite/doors>.
- 49 Jonas Rauber, Roland Zimmermann, Matthias Bethge, and Wieland Brendel. Foolbox native: Fast adversarial attacks to benchmark the robustness of machine learning models in PyTorch, TensorFlow, and JAX. *Journal of Open Source Software*, 5(53):2607, 2020. doi:10.21105/joss.02607.
- 50 SAE. Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles. SAE J3016, 2021. URL: https://www.sae.org/standards/content/j3016_202104/.
- 51 Wojciech Samek et al. *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Springer, 2019. doi:10.1007/978-3-030-28954-6.
- 52 Wojciech Samek, Grégoire Montavon, Andrea Vedaldi, Lars Kai Hansen, and Klaus-Robert Müller. *Explainable AI: interpreting, explaining and visualizing deep learning*, volume 11700. Springer Nature, 2019.
- 53 Anna Schmitz, Michael Mock, Rebekka Görges, Armin B Cremers, and Maximilian Poretschkin. A global scale comparison of risk aggregation in AI assessment frameworks. *AI and Ethics*, pages 1–26, 2024.
- 54 Seldon. Alibi Explain. URL: <https://github.com/SeldonIO/alibi>.
- 55 sicara. TF-Explain: Interpretability Methods for tf.keras Models with TensorFlow 2.x. URL: <https://github.com/sicara/tf-explain>.
- 56 Siemens. Polarion, 2004. URL: <https://polarion.plm.automation.siemens.com/>.
- 57 Ryan Soklaski, Justin Goodwin, Olivia Brown, Michael Yee, and Jason Matterer. Tools and practices for responsible AI engineering. *arXiv preprint arXiv:2201.05647*, 2022. arXiv: 2201.05647.
- 58 Guillaume Soudain. First usable guidance for Level 1 machine learning applications: A deliverable of the EASA AI Roadmap, 2021. URL: <https://www.easa.europa.eu/en/downloads/134357/en>.
- 59 The White House. Fact Sheet: President Biden Issues Executive Order on Safe, Secure, and Trustworthy Artificial Intelligence, 2023. URL: <https://www.whitehouse.gov/briefing-room/statements-releases/2023/10/30/fact-sheet-president-biden-issues-executive-order-on-safe-secure-and-trustworthy-artificial-intelligence/>.
- 60 Trusted-AI LF AI Foundation. Adversarial Robustness Toolbox (ART). URL: <https://github.com/Trusted-AI/adversarial-robustness-toolbox>.
- 61 Trusted-AI LF AI Foundation. AI Explainability 360 (AIX360). URL: <https://github.com/Trusted-AI/AIX360>.
- 62 Trusted-AI LF AI Foundation. AI Uncertainty Quantification 360 (UQ360). URL: <https://github.com/Trusted-AI/UQ360>.
- 63 Arnaud Van Looveren et al. Alibi detect: Algorithms for outlier, adversarial and drift detection, 2019. URL: <https://github.com/SeldonIO/alibi-detect>.
- 64 Larysa Visengeriyeva, Anja Kammer, Isabel Bär, Alexander Kniesz, and Michael Plöd. MLOps Principles, 2020. URL: <https://ml-ops.org/content/mlops-principles>.

Original Paper

Finding Consensus on Trust in AI in Health Care: Recommendations From a Panel of International Experts

Georg Starke^{1,2}, MD, PhD; Felix Gille^{3,4}, PhD; Alberto Termine^{1,2,5}, PhD; Yves Saint James Aquino⁶, MD, PhD; Ricardo Chavarriaga⁷, PhD; Andrea Ferrario⁸, PhD; Janna Hastings^{4,9}, PhD; Karin Jongsma¹⁰, PhD; Philipp Kellmeyer^{11,12}, MD; Bogdan Kulynych¹³, PhD; Emily Postan¹⁴, PhD; Elise Racine^{15,16}, MPA, MSc; Derya Sahin¹⁷, MD; Paulina Tomaszewska¹⁸, MSc; Karina Vold^{19,20}, PhD; Jamie Webb²¹, PhD; Alessandro Facchini⁵, PhD; Marcello Ienca^{1,2}, PhD

¹Institute for History and Ethics of Medicine, Technical University of Munich, Munich, Germany

²College of Humanities, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland

³Digital Society Initiative, University of Zurich, Zurich, Switzerland

⁴Institute for Implementation Science in Health Care, Faculty of Medicine, University of Zurich, Zurich, Switzerland

⁵Dalle Molle Institute for Artificial Intelligence (IDSIA), The University of Applied Sciences and Arts of Southern Switzerland (SUPSI), Lugano, Switzerland

⁶Australian Centre for Health Engagement, Evidence and Values, University of Wollongong, Wollongong, Australia

⁷Centre for Artificial Intelligence, Zurich University of Applied Sciences (ZHAW), Zurich, Switzerland

⁸Institute of Biomedical Ethics and History of Medicine, University of Zurich, Zurich, Switzerland

⁹School of Medicine, University of St. Gallen, St. Gallen, Switzerland

¹⁰Bioethics & Health Humanities, University Medical Center Utrecht, Utrecht University, Utrecht, The Netherlands

¹¹Data and Web Science Group, School of Business Informatics and Mathematics, University of Mannheim, Mannheim, Germany

¹²Department of Neurosurgery, University of Freiburg - Medical Center, Freiburg im Breisgau, Germany

¹³Lausanne University Hospital (CHUV), Lausanne, Switzerland

¹⁴Edinburgh Law School, University of Edinburgh, Edinburgh, United Kingdom

¹⁵The Ethox Centre and Wellcome Centre for Ethics and Humanities, Nuffield Department of Population Health, University of Oxford, Oxford, United Kingdom

¹⁶The Institute for Ethics in AI, Faculty of Philosophy, University of Oxford, Oxford, United Kingdom

¹⁷Development Economics (DEC), World Bank Group, Washington, DC, United States

¹⁸Faculty of Mathematics and Information Science, Warsaw University of Technology, Warsaw, Poland

¹⁹Institute for the History and Philosophy of Science and Technology, University of Toronto, Toronto, ON, Canada

²⁰Schwartz Reisman Institute for Technology and Society, University of Toronto, Toronto, ON, Canada

²¹The Centre for Technomoral Futures, University of Edinburgh, Edinburgh, United Kingdom

Corresponding Author:

Georg Starke, MD, PhD

Institute for History and Ethics of Medicine

Technical University of Munich

Ismaninger Str. 22

Munich, 81675

Germany

Phone: 49 8941404041

Email: georg.starke@tum.de

Abstract

Background: The integration of artificial intelligence (AI) into health care has become a crucial element in the digital transformation of health systems worldwide. Despite the potential benefits across diverse medical domains, a significant barrier to the successful adoption of AI systems in health care applications remains the prevailing low user trust in these technologies. Crucially, this challenge is exacerbated by the lack of consensus among experts from different disciplines on the definition of trust in AI within the health care sector.

Objective: We aimed to provide the first consensus-based analysis of trust in AI in health care based on an interdisciplinary panel of experts from different domains. Our findings can be used to address the problem of defining trust in AI in health care applications, fostering the discussion of concrete real-world health care scenarios in which humans interact with AI systems explicitly.

Methods: We used a combination of framework analysis and a 3-step consensus process involving 18 international experts from the fields of computer science, medicine, philosophy of technology, ethics, and social sciences. Our process consisted of a synchronous phase during an expert workshop where we discussed the notion of trust in AI in health care applications, defined an initial framework of important elements of trust to guide our analysis, and agreed on 5 case studies. This was followed by a 2-step iterative, asynchronous process in which the authors further developed, discussed, and refined notions of trust with respect to these specific cases.

Results: Our consensus process identified key contextual factors of trust, namely, an AI system's environment, the actors involved, and framing factors, and analyzed causes and effects of trust in AI in health care. Our findings revealed that certain factors were applicable across all discussed cases yet also pointed to the need for a fine-grained, multidisciplinary analysis bridging human-centered and technology-centered approaches. While regulatory boundaries and technological design features are critical to successful AI implementation in health care, ultimately, communication and positive lived experiences with AI systems will be at the forefront of user trust. Our expert consensus allowed us to formulate concrete recommendations for future research on trust in AI in health care applications.

Conclusions: This paper advocates for a more refined and nuanced conceptual understanding of trust in the context of AI in health care. By synthesizing insights into commonalities and differences among specific case studies, this paper establishes a foundational basis for future debates and discussions on trusting AI in health care.

(*J Med Internet Res* 2025;27:e56306) doi: [10.2196/56306](https://doi.org/10.2196/56306)

KEYWORDS

expert consensus; trust; artificial intelligence; clinical decision support; assistive technologies; public health surveillance; framework analysis

Introduction

Background

The integration of artificial intelligence (AI) systems into health care is one of the most widely anticipated transformations of health systems worldwide [1]. AI promises improved diagnostics [2,3], optimized treatment strategies [4], and early identification of at-risk patients [5]. Prominent examples also include AI for assistive technologies offered to patients directly [6,7] and for informing public health decision-making beyond the individual [8]. More recently, large language model-based applications promise to revolutionize health care, with applications spanning from clinical research and processes to physician-patient relations [9]. Despite this range of potentially beneficial applications, the broader adoption of AI systems in health care has been struggling due to many inhibiting factors. Problems arise at the level of development, with data bottlenecks impeding the training of machine learning models or a lack of user-centered and value-sensitive design procedures affecting AI acceptability [10,11], and stretch to the level of practical implementation of AI systems [12-14]. Questionable improvements in real-life health care settings [15], a lack of regulatory frameworks [16], and unresolved questions of reimbursement [17] further complicate adoption of AI in health care.

In this paper, we focus on a central inhibitor of successful AI adoption in health care, namely, the low levels of user trust in AI systems [18]. Understanding and fostering trust in AI remains challenging, not only practically but also conceptually. Interpersonal trust constitutes a complex and contested construct

in philosophy and social sciences [19-24]. Moreover, accounts diverge on what constitutes being worthy of trust, namely, on the definition of *trustworthiness* [19,20]. Unsurprisingly, there is also no consensus across disciplines such as computer science, philosophy of technology, and social sciences on the definition of trust in AI and the capabilities that a trustworthy AI should maintain [25,26]. As a recent publication assembling existing empirical literature on trust in AI and clinical decision support put it, “Different groups [of researchers], whilst seemingly agreeing in principle that ‘trust’ and ‘trustworthiness’ are important, can in fact be referring to very different concepts and talking past one another” [27].

The combination of fragmented conceptual research, practical concerns, and implementation difficulties inhibits fostering warranted trust in AI (ie, trust caused by the trustworthiness of the AI) [25,28], which will be increasingly crucial for delivering health care [29]. As recent empirical work using path analysis has highlighted, trust in AI-based technologies seems to have a significant effect on users' intention to use such systems [30]. However, to build trust in—possibly trustworthy—AI systems in health care applications, we first need a conceptual understanding of trust in AI within the health care sector. This goal requires considering the diverse perspectives, expectations, and limitations that various research disciplines bring to the table in the context of human-AI interactions. Without conceptual clarity, attempts to foster the multifaceted concept of trust in AI—whatever it may *precisely* mean—run the risk of being inefficient or even detrimental, leaving users, including data scientists, physicians, and patients, vulnerable to placing trust in systems that may not warrant it or refraining from it in

situations in which grounds for trusting actually exist, an occurrence of what we might call “unwarranted distrust” [31,32].

Objectives

Bringing together international and multidisciplinary perspectives on the topic of trust in AI, this paper aims to provide common ground for defining trust in AI in health care as a reference point for future debates. To this end, we developed the first consensus statement on trust in AI in health care based on input from international experts on the topic drawing on iterative synchronous and asynchronous discussions of realistic case studies. Our results highlight the need for a more refined and nuanced understanding of trust in the context of AI in health care if the concept is to guide AI design and adoption processes and inform national and international governance.

Methods

Overview

Consensus statements from meetings of experts working on a particular topic constitute a common scientific approach across the fields of medicine [33,34] and ethics [35,36]. While they can take many different forms [37], their shared goal is to identify agreement among people working in a field and assemble multiple perspectives with peer-informed legitimacy

[38]. The reporting of our findings follows the Accurate Consensus Reporting Document guidelines [37] and considered applicable elements of guidelines for reporting qualitative research [39,40].

Recruitment

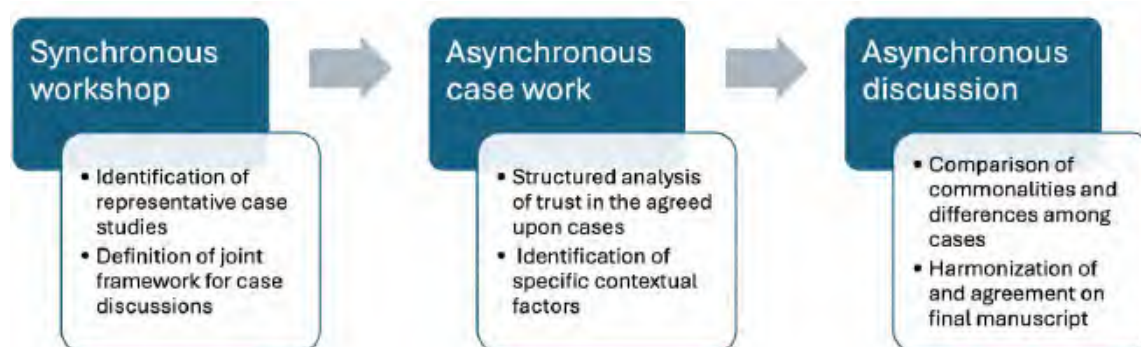
Recruitment for the consensus process took place as part of a 3-day workshop held at the École Polytechnique Fédérale de Lausanne in September 2022 in Lausanne, Switzerland. Authors GS, FG, AT, A Facchini, and MI, who organized the workshop, invited some participants directly based on their contribution to the field and to accommodate different types of expertise and backgrounds. They selected further participants from submissions to a call for abstracts for the workshop. Unanimous agreement concerning participants was achieved among conference organizers GS, FG, AT, A Facchini, and MI.

Consensus Process

Overview

The consensus process comprised 3 sequential steps involving both synchronous (step 1) and asynchronous (steps 2 and 3) activities. We describe them in the following sections. Figure 1 shows the steps of the consensus process and provides an overview of the intended outputs of each step.

Figure 1. The 3-step consensus process and envisioned results.



Synchronous Phase: Selection of Case Studies and Framework

We conducted a workshop involving an international group of researchers from various disciplines. To arrive at a consensus regarding trust in AI in health care, we discussed the topic from complementary angles in a series of group activities that iteratively involved talks on specific aspects of trust, moderated group discussions, plenary sessions, and interactive panels. The moderated group discussions were spread over 3 days. They covered the topics of (1) conceptualizing trust in the context of human-AI interactions, (2) conceptualizing trust in AI in the context of health care specifically, (3) requirements for trustworthy AI in health care, and (4) recommendations for implementing requirements for trustworthy AI in health care. To achieve consensus on the case studies, all participants reviewed them. Only case studies assessed as representative yet distinct from other cases by all participants were promoted to the next step.

To guide our analysis, we made use of a conceptual framework. The framework was proposed by FG, critically discussed, and agreed upon by all authors. A conceptual framework represents a phenomenon in the form of a network with interlinked themes that describe how the phenomenon works [41]. Conceptual framework analysis provides an established qualitative method to guide the identification and organization of key aspects and relationships in contexts that draw on multidisciplinary bodies of knowledge [41]. We used framework analysis to provide structure and coherence of discussions for the asynchronous steps, systematically evaluating various dimensions of trust in AI in health care.

Asynchronous Step: Developing Case Studies on Trust in AI in Health Care

To understand how trust unfolds in the context of the agreed upon 5 cases, in the second step, smaller groups of authors expanded on the individual cases agreed upon in step 1. All contributors involved in the writing are listed as authors.

Participants in this step were asked to explicate trust-relevant factors of the cases in accordance with the agreed upon conceptual framework. Responsibilities for coordinating the writing of individual case studies were distributed among the authors, with 1 person taking responsibility for leading each case study. KJ led the consensus for the first case study, JH led the consensus for the second case study, YSJA led the consensus for the third case study, GS led the consensus for the fourth case study, and JW led the consensus for the fifth case study. The results were compiled and shared by GS, FG, and MI.

Asynchronous Step: Comparison and Discussion of Case Studies

Having combined the individual case studies, we compared and discussed relevant aspects to synthesize our findings. We discussed our findings critically in light of existing literature and asked participants again to provide written feedback on the synthesized findings. This process was coordinated by GS, who wrote a first draft of the manuscript, which was then refined by FG, A Facchini, A Ferrario, and MI. All authors provided their feedback on the manuscript, which was incorporated and harmonized by GS and shared with all authors for approval.

Ethical Considerations

In accordance with the existing regulatory framework in Switzerland, such as the Human Research Act and the local rules at the École Polytechnique Fédérale de Lausanne as hosting institution, no ethics approval was required for this collaborative research project involving only consenting experts in the field.

Results

Overview

The results of our study unveil the complexities of understanding trust in AI within the health care domain. Our findings represent consensus among international experts guided by conceptual framework analysis. The following sections provide the characteristics of the involved experts and report the findings from the individual synchronous and asynchronous steps of our consensus process.

Recruitment

The characteristics of the 18 participants in the consensus process are described in [Table 1](#).

Table 1. Participant characteristics—multiple entries possible for location and background (N=18).

Characteristics	Participants, n (%)
Gender	
Women	7 (39)
Men	11 (61)
Nonbinary	0 (0)
Prefer not to say	0 (0)
Geographic location	
Europe	16 (89)
North America	2 (11)
Oceania	1 (6)
Background	
Computer science	6 (33)
Ethics	5 (28)
Medicine	4 (22)
Philosophy	8 (44)
Public health	2 (11)
Social science	3 (17)

Consensus on Trust and Trustworthiness

Synchronous Phase: Selection of Case Studies and Framework

A total of 5 case studies emerged from the synchronous group discussion and were identified as paradigmatic examples of the various applications of AI in health care, namely, diagnosis, clinical risk assessment, public health surveillance, assistive technologies, and health care resource allocation. The involved experts agreed that these cases may not provide an exhaustive sample of all potential AI applications in health care yet allow

for a meaningful representation and comparison of trust in AI across largely different health care settings.

To guide the asynchronous analysis of the individual cases, all involved experts identified and defined key components of a conceptual framework for trust in AI in health care. We developed this framework with the backdrop of existing conceptual work on trust in data use in health care, preliminary discussions of the shortcomings of present conceptual work describing what trust in AI is, and existing guidance on conceptual framework development for research and scale development in medicine [29,31,42-49].

Following existing conceptualizations of trust in the context of data use in health care [47], themes were grouped into two main areas: (1) the context of trust and (2) causes and effects of trusting. We considered trust as a context-specific, relational construct between 2 actors that is shaped by the environment in which it develops over time [50]. Therefore, our framework implemented the context specificity of trust by considering (1) the environment within which an AI system is deployed (ie, the health care application in which the AI operates), (2) the actors involved in the specific trust relationship, and (3) the factors that frame the trusting relationship. Here, framing factors influence “the process by which people develop a particular conceptualization of an issue or reorient their thinking about an

issue” [51]. Examples of framing factors are historical context, cultural aspects, norms and values, fears, public sentiment, overarching belief systems, and religious beliefs. In addition to context, our framework includes factors that support or inhibit trust as well as the effects of trusting AI systems. Causally important themes for trusting relationships typically relate to factors that make AI trustworthy or untrustworthy in the eyes of those placing trust. Examples are the reliability and accuracy of the AI and, arguably, the level of interpretability of its outcomes [28,52]. In summary, our framework comprised 5 themes: environment, actors, frames, causes, and effects (Textbox 1).

Textbox 1. Conceptual framework for warranted trust in artificial intelligence (AI) in health care.

Context of trust in AI in health care

- Environment: What is the setting for which the AI system is intended?
- Actors: Who are the involved actors in the trust relationship?
- Frame: What frames trust in relation to AI?

Cause and effect of trust in AI in health care

- Cause: What makes AI trustworthy?
- Effect: What is the effect of trust in relation to AI?

Asynchronous Step: Developing Case Studies on Trust in AI in Health Care

Drawing on our framework and previous research, smaller groups of authors examined particular aspects of AI used for diagnostic purposes, clinical risk assessment, public health surveillance, assistive technologies, and health care resource allocation. The results of the individual groups led by KJ, JH, YSJA, GS, and JW, respectively, are reported in the following sections.

Diagnostic AI in the Clinic

Machine learning and, more specifically, deep learning have proven to be particularly suitable for computer vision, especially image processing and pattern recognition. Important preconditions to apply these techniques in medicine, such as suitable infrastructure or availability and storage of digital images, are also being increasingly met, at least in high-income countries. This may explain why most current and proposed AI systems in medicine are used to aid image-based diagnostics in fields such as radiology, ophthalmology, and pathology [53-55]. As a large part of the workload of, for instance, radiologists is to interpret medical images [56], diagnostic AI systems could increase this capacity; support or take over the tasks; and, thereby, change the organization of image-driven diagnostics [57].

Some studies have indicated that, under specific circumstances, AI systems achieve accuracy that is at least equal to that of expert radiologists and pathologists or even outperform them when detecting, classifying, and segmenting tumors in ultrasonography, x-ray imaging, magnetic resonance imaging scans, and digitized microscopy slides [58,59]. These findings have raised massive enthusiasm and fueled the motivation to

develop and use algorithms in image-driven diagnostics. As a particular example, we will focus on a recently approved AI system for the diagnostic analysis of chest x-rays [60].

The system aids diagnosis by analyzing chest x-rays, the most frequent radiological examination worldwide. It does so by classifying the scans in nonpathological and potentially pathological cases. For the first group, the program provides a fully automated report, removing the necessity for any further follow-up on the image by radiologists. The scans of the second group are forwarded to trained physicians for further radiological analysis. A test of the software on approximately 10,000 chest x-rays from Finland showed a very high sensitivity (99.8%) with a low specificity (36.4%), yielding a very low probability of missing any critical findings [61].

However, the use of diagnostic AI is associated with technological obstacles as well as epistemic and ethical challenges [62]. Although the potential of self-supervised learning to generate expertlike annotations has been demonstrated, an insufficient number of expertly annotated diverse images is currently often a limiting factor in training AI models on a technical level [63]. Large datasets, for instance, in pathology, can also pose problems if downsampling leads to crucial information being lost, demanding different kinds of representation [64]. Beyond such technical difficulties, there are also more fundamental obstacles, including the fact that diagnostic tasks are not usually choices between 2 distinct outcomes [65]. Another concern is the frequent lack of explainability of medical AI, which limits physicians' ability to recheck the AI's output [66]. It has been argued that this raises particular challenges as decisions based on black-box AI systems are potentially not fully interpretable for the human physician. Moreover, when offering explanations, it is essential

to consider the epistemic requirements of users (eg, radiologists and patients). Explanations that lack a clear connection to clinical knowledge despite being technically accurate fall short in delivering the desired support for clinical practice. Similarly, patients who wish to understand the basis of their diagnosis may find explanations that diverge from their own illness experiences unsatisfactorily [67,68]. That said, there is empirical evidence that a lack of explainability is not necessarily perceived as a problem by physicians if a diagnostic system has been properly validated and other detailed information about the AI system is available [69,70].

The case discussed here takes place in a clinical environment of image-driven diagnostics. It directly involves medical professionals and the AI system, whereas patients do not interact with the system itself; in addition, the developing company and regulatory bodies are indirectly involved. Framing factors are discourses about job security in radiology and the danger of physicians being replaced by AI [71]. Trust leads to acceptance of the system by physicians, potentially at the cost of deskilling, and is arguably fostered by the accuracy of the diagnostic program in question [72].

Predictive AI for Clinical Risk Assessment

AI algorithms can be used to predict individual patient trajectories. Examples include the prediction of COVID-19 severity [73], the prediction of delirium [74], prognostic models of respiratory diseases [75], or systems predicting future lung cancer risk [76]. In addition, predictive models are crucial for determining individualized treatment plans, often particularly relevant in oncology [77]. The difference from the diagnostic use case is that the prognostic case concerns the future (ie, informs decision-making under situations characterized by an inherently larger uncertainty). The prognostic prediction informs choices of action, treatments offered, support, and resource allocation and, therefore, can, in turn, influence the very outcome that has been predicted [78].

As a specific case study, let us consider the prediction of circulatory failure in intensive care units (ICUs) [5]. In ICUs, a great number of machines monitor the state of patients and need to be observed constantly as patient conditions may worsen rapidly. The existing technologies for life support and vital sign monitoring provide information and alerts that clinicians need to oversee. However, the high rates of alerts, including false positives, may cause alert fatigue for practicing clinicians, impairing optimal care. In this context, systems that predict severe deterioration more accurately can obviate alert fatigue and potentially lead to better patient outcomes. The system reported in the study by Hyland et al [5] integrates information from multiple organ function-monitoring systems to alert clinicians to potential circulatory failure 8 hours in advance. Note that the information available to the system is necessarily incomplete, and it is impossible to predict future circulatory failure with 100% accuracy using any algorithm. However, the system in question successfully predicted 90% of cases of circulatory failure in the test set and 82% of them >2 hours before the event [5], so it would likely prove useful in a clinical setting.

The system described here can only be deployed in the environment of hospitals or clinics. Physicians, nurses, and potentially caregivers may interact directly with the AI system, and the social circle of a patient could also potentially receive information derived by the AI system, whereas the patients admitted to an ICU will likely not interact with the system itself. Critical contextual factors are the system's use in stressful situations with the risk of severe consequences, sometimes under time pressure, and the need to synthesize a multitude of complex information. Trust in such a situation can be built by overall improved outcomes in settings where the algorithm is used, ideally by investigating it through a randomized controlled trial. In addition, there should be reasons to believe that the AI system is not biased or will not lead to harmful consequences for any subgroups of the population and that it produces reliable individual decisions that are robust to design choices [79,80] or randomness [79,81] in the AI pipelines. Trust built on these premises will promote AI's acceptance and use. However, as with other medical interventions aiming to change the future, such as screening programs [82,83], trust could also lead to worse outcomes if the system does not actually deserve it (ie, if it is not *trustworthy*).

AI for Public Health Surveillance

Public health surveillance involves the identification of signs of population-level health anomalies and potential disease outbreaks from a heterogeneous collection of data sources [8]. With its intrinsically data-driven nature, public health surveillance has increasingly become a domain for AI applications as AI provides a range of novel methods for data collection and data analysis in large and varied samples [8]. A comprehensive survey of the literature showed that the COVID-19 pandemic highlighted the potential of AI systems in improving surveillance of infectious disease outbreaks [84].

Consider the example of EPIWATCH among the numerous AI-based surveillance systems that provide early signs of disease outbreaks [85]. Developed at the University of New South Wales in Australia, EPIWATCH is an open access web-based tool that offers an interactive dashboard with a sortable, searchable, and filterable global map based on 30 days of data [85]. The tool has been successfully used to model and identify patterns of disease outbreaks, providing crucial information on risk factors and geographic distribution.

For instance, using publicly available data, EPIWATCH has been used to trace global Zika virus outbreaks, tracing transmission modes, affected countries, and complications such as microcephaly [86]. Similarly, a different group of authors used EPIWATCH to model the global epidemiology of hepatitis A, identifying the United States and Europe as major centers of outbreaks, and provided quantitative global data on the most common risk factors, which seem to be homelessness and foodborne outbreaks [87]. Such data can be helpful to inform policy making at both a global and local level.

Let us summarize the contextual factors of this case. Contrary to the previous examples, AI systems for public health are deployed in nonclinical settings. This web-based example is potentially accessible by anyone with a computer or mobile phone with an internet connection interested in data analysis to

predict disease outbreaks. Therefore, relevant actors include the developers as well as a broad spectrum of potential users, ranging from public health practitioners and policy makers to the public. Important framing factors for trust are the severity of the modeled disease as well as the stage of the epidemic or pandemic, impacting the relevance and acceptability of the tool. As the COVID-19 pandemic has demonstrated, antiscience sentiments and conspiracy theories contribute to how public health tools are viewed. Finally, usability aspects further influence whether information will be taken up by practitioners. Trust in the application will likely lead to increased uptake among decision makers. The trustworthiness of the platform developers and endorsement by authorities, such as public health experts and policy makers, will further determine whether the wider public will trust the model's output.

AI for Assistive Neurotechnology

In comparison to the examples discussed previously, a quite different trust relationship can be found in the case of assistive neurotechnologies based on brain-computer interfaces (BCIs). BCIs can, for instance, be used for neurorehabilitation, restoring lost function, or augmenting and enhancing existing cognitive or motor abilities [7]. AI, especially data-driven machine learning with deep neural networks, plays a crucial role in their development and use as these methods enable and facilitate the modeling and decoding of complex neural signals as well as the adaptation to individual users [88]. To delineate differences in the trust relationship between AI-based neurotechnology and other medical AI, we focus on a specific example: a BCI-guided robotic hand orthosis for rehabilitative purposes.

Orthoses are external wearable devices that can be “used to compensate for impairments of the structure and function of the neuromuscular and skeletal systems” [89]. In line with numerous existing research projects at the level of preclinical prototypes [90], let us consider a specific BCI-based robotic orthosis that allows patients with stroke to restore some volitional control of hand-grasping movements. To do so, the system records electrical activity of the dominant motor cortex noninvasively through an electroencephalography cap and extracts features from the measured signals using machine learning-based classifiers to extract patient-specific neural markers of intended hand movements, which are then translated into mechanical movements of the robotic hand orthosis.

There are several aspects of such personalized AI-based assistive neurotechnologies for neurorehabilitation that set them apart from medical AI used for diagnosis, prediction, or public health surveillance. One crucial difference that stands out is that the AI in question is *embodied* in the sense that the AI used for decoding neural signals and for translating them to mechanical movement is inextricably linked to a physical object—namely, the hand orthosis. This implies that physical design aspects play a crucial role on whether trust is expedited to the device [91].

However, there are further differences that impact the trust relationship involving actors, the environment, and framing. First, assistive rehabilitative neurotechnology is designed for patients with impairment of cognitive or motor functions who need to engage with the technology. These patients are also the main trusters of this technology, not health care professionals

or public health experts. Second, suppose the neurotechnology also helps patients in their everyday life. In that case, it should not only work under highly constrained and controlled laboratory or clinical conditions but also in a patient's home. This implies additional demands for the devices' ease of use and their security in a more open environment, which would be crucial for the trust of bystanders in these devices. Third, due to the BCIs' physicality, trust in assistive neurotechnology may also be framed differently, invoking widely known images from science fiction [92]. Fourth, users may be concerned that decoding their brain activity could be used not only to control the robotic orthosis but also to infer other types of information that would not be available using other sensing technologies, thus raising potential threats to their privacy. Therefore, addressing potential fears of patients and designing AI that assures human control (in the sense of “human in the loop”) at any moment adds to the complexities of building trust in such embodied AI.

To summarize, the assistive AI system discussed here is embedded in an environment of clinical neurorehabilitation. Its trusters are primarily patients and rehabilitation experts who trust in the developing engineers and the pertinent regulatory bodies. In the context of embodied AI, many specific framing factors influence its perception, especially from science fiction literature and cinema. It is also shaped by public attitudes and policies on related technologies such as robotics. To build trust in this environment and prove the accuracy of the AI, a lack of conflicts of interest on the part of the developers; sufficient understanding of the underlying technology by its users and rehabilitation experts; and independent, long-term technical support seem key. Successful trust building will result in wider acceptance and adherence by users, as well as potential inclusion of the technology in public health programs, facilitated regulatory compliance, and improved insurance reimbursement. Taken together, these elements can facilitate access and increase the technology's affordability by expanding the number of its users.

AI for Health Care Resource Allocation

A 2019 study in *Science* revealed that software provided by the health service company Optum, and which was being used to manage the care of >200 million patients at hospital centers across the United States, was significantly—even if unintentionally—biased against Black patients [93]. The machine learning algorithm aimed to predict the future health care needs of patients and direct extra medical care toward the most vulnerable. However, it was shown to systematically underestimate the needs of Black patients. It did this because it used health care costs as a proxy for need in its risk score despite health care costs not being a neutral and reliable proxy for health needs in this context. This led it to assign consistently lower risk scores to Black patients compared to White patients who were equivalently sick. This was despite the model being “race blind” in the sense that race was not specified in the input data [94].

When considering how this case study relates to trust in and trustworthiness of AI in health care, it is important to note that this is not a case of AI being introduced into a dyadic

physician-patient relationship but, rather, of its integration into the operations of complex health systems involving many actors and institutions. Therefore, philosophical accounts of trust that focus on the properties of individuals—these accounts traditionally comprise reliability *plus* some appropriate motivational state on the part of the trustee toward the truster, for example, goodwill [95]—are insufficient to capture the conditions necessary for trustworthy AI in these contexts.

Accounts of institutional trustworthiness are more useful here. They often emphasize features such as transparency as well as competence and reliability [96]. Accounts of trustworthiness that do not necessarily require a phenomenological state such as goodwill may also be useful. For example, the trust responsiveness account by McGeer and Pettit [97] emphasizes the importance of an agent—or, in our case, institutions—responding appropriately to the reasons for doing what they are being relied upon to do. Alternatively, one might provide a deflationary account that either equates trustworthiness with reliability [48] or rejects trust as an appropriate attitude altogether [98].

What does it mean to meet these conditions in this and similar cases of integrating machine learning into resource allocation decisions across an entire hospital or health system? Trust responsiveness demands that systems such as the one provided by Optum be introduced to promote health and well-being and improve patient outcomes and not to meet other goals such as cutting costs to maximize profits for insurance companies and health systems. Using a system such as Optum may not necessarily require internal algorithmic transparency, which is often the focus of work considering the ethics of machine learning algorithms in health care [99], yet may be difficult to translate into a genuine understanding of the AI system by clinicians or patients. However, there needs to be openness regarding design decisions and value choices involved in the algorithm's construction and implementation [100]. Demonstrating these may require engaging in ethical and algorithmic impact assessments during the design process [101,102], welcoming algorithmic auditing after implementation [49], and abiding by regulatory frameworks throughout the product life cycle [103]. Ideally, transparency for AI systems for resource allocation would highlight both global explanations [104], given that the goal of the algorithm is to distribute resources across a population [105], and local explanations for individual decisions as the biased program impacts individual care. Both are potentially useful for promoting fairness [104].

In addition, it is crucial to consider contextual factors impacting such trust relationships. The program was developed by a for-profit health service provider in the US health care system and used to determine resource allocation. Key actors were the health service company itself, which provided the algorithm; the health systems; clinicians interacting with the algorithm's recommendations; patients having their care influenced by the algorithm's recommendations; and, finally, regulatory bodies and (internal and external) algorithmic auditors. Important

framing factors include media reporting on algorithms and their impact and, on a conceptual level, theories of institutional trust. Such trust, though unwarranted here, would result in the model's acceptance and use. It is supported by an AI's reliability and competence as well as by the transparency of the developers concerning design choices and the values reflected in them.

Asynchronous Step: Comparison and Discussion of Case Studies

As the 5 cases highlight, medical AI is a multifaceted concept encompassing a diverse group of systems, environments, actors, and framing factors. Talking about trust in medical AI in general can only be a first approximation to the phenomenon. However, some general elements of trust were accepted by all experts. First, they agreed that trust provides a way to deal with complex situations in the face of uncertainty [106,107] and is established in anticipation of a beneficial outcome. Therefore, in a rather minimal fashion, and in line with existing literature, an agreed upon definition of trust in AI in health care focuses on giving discretionary power to an AI system with respect to a specific health care–related task [29,108]. At the same time, it was agreed that situations of trust were characterized by risks, rendering the trusting agent vulnerable to betrayal of trust [21]. To minimize such risks, potentially of life and death in health care, experts agreed that we need to only promote warranted trust in medical AI (ie, trust that is justified, plausible, and well grounded [19,28,49]). In the literature as well as in regulatory contexts, AI systems that warrant such trust have been described as trustworthy [49,109,110]. Therefore, in the eyes of the experts, a complete assessment of trust in AI includes epistemological and practical considerations of trust as much as examining the trustworthiness of the AI in question (ie, the functionalities of an AI system that morally and practically vindicate a relationship of trust [111]).

The involved experts agreed on many practical shortfalls of current AI in health care potentially impacting its trustworthiness. These include the lack of explainability of many AI algorithms [28,66,112]; the difficulties of benchmarking model performance [113]; or the longtail of exceptional situations that will be difficult to anticipate at the development stage, from nonstandard data inputs to the system's deployment in a health care system for which it was not originally trained [114]. In the eyes of the experts, anticipatory difficulty also relates to risks of AI perpetuating, hiding, or reinforcing statistical and social biases as well as other data problems [115,116]. Finally, it was stressed that many medical AI tools so far lack proper engagement with stakeholders in their development, which is crucial for ensuring that they adequately address a real clinical problem and identifying their relevant ethical implications [117].

When looking at the individual case studies, commonalities and differences among them arose at micro and macro levels. Table 2 provides an overview of the contextual factors of each case study.

Table 2. Contextual factors for the 5 case studies.

	Environment	Actors	Framing factors	Causes of trust	Effects of trust
Diagnostic AI ^a (chest x-rays)	Image-driven diagnostics (radiology)	Medical professionals and AI system; patients to a limited extent	Discourse regarding job security and potential AI replacement	Accuracy, design transparency, and human competencies and virtues	Acceptance of systems by physicians, potentially at the cost of deskilling
Predictive AI (ICU ^b setting)	Clinical setting of an ICU	Physicians, nurses, and AI system; patients to a limited extent; and potentially caregivers and family members	Stressful situations potentially with a need to act under time pressure, risk of severe consequences, the need to synthesize too much information, and alert fatigue	Accuracy, transparency, and explainability; fairness; exclusion of harm; and rigorous testing (eg, in the form of an RCT ^c)	Acceptance and use of the system, potentially at the risk of erroneous clinical decisions following misleading predictions
Public health AI (disease outbreak model)	Nonclinical setting—publicly accessible web-based tool for the analysis of heterogeneous data	Developers, public health practitioners, policy makers, and the public	Stage and severity of disease outbreak; usability aspects (eg, intuitive interface or data visualization); and, potentially, antiscience sentiments and conspiracy theories with regard to the disease and health service providers	Historical accuracy and endorsement by authorities	Acceptance and use of the system by public decision makers (public health experts and policy makers)
Assistive AI (neurorehabilitation)	Clinical neurorehabilitation; elective use of different technologies for different activities, potentially every day	Patients and their caregivers and social circle, potentially including employers, engineers, and regulators	Clinical setting, science fiction literature and cinema, and public attitudes and policies on related technologies	Accuracy, privacy, lack of conflicts of interest, independence, long-term technical support, and user understanding of the underlying technology	Technology acceptance by users, health care professionals, and health care providers; potentially facilitated reimbursement and increased affordability and accessibility
Resource-allocating AI (predicting costs and needs)	Health service providers and health care system	The developing company providing the algorithm, the health system implementing it, the clinicians interacting with it, the patients having their care influenced by the algorithm, and regulatory bodies and algorithmic auditors	Media reporting on algorithms and their impact and theories of institutional trust	Reliability, accuracy, transparency, design, and model-centric explanations	Acceptance and use in health care systems

^aAI: artificial intelligence.^bICU: intensive care unit.^cRCT: randomized controlled trial.

Most commonalities among the different cases can be found with regard to their causes and effects, some of which arise from the very structure of trust. A common effect of trust was found in it leading to wider acceptance of the technology in question. Such acceptance implies an increased uptake by its intended end users, resulting in a larger role in health care systems, facilitating access, and potentially increasing the systems' affordability. In line with the literature, common aspects with regard to trust-supporting features included both aspects intrinsic to the system, such as transparency and explicability, and extrinsic factors, such as proper external validation and assessment of potential biases [25,29].

Across the discussed cases, some forms of transparency were identified as supportive of establishing trust. Such transparency included openness by the developers about their measures of a well-working system, enabling an alignment with the goals and values of patients and clinicians. Promoting transparency was also assessed as a good way of promoting trust among clinicians

using the systems. For instance, there is some empirical evidence that being transparent about the functioning of a triage system makes clinicians more likely to trust it [118]. The extent to which this is possible depends on the exact nature of the system, with clear differences among algorithms that are interpretable by design [119], algorithms that can be explained in terms of their exact functioning on the level of predictors and their weights, and algorithms whose decision process cannot be interpreted causally.

Informational openness was also recognized as central to promoting a system's external trustworthiness (eg, if a system's reliability and performance metrics are publicly available). From the discussed cases, it also emerged that rendering information accessible to all relevant auditing bodies is crucial for mechanisms of accountability and, in doing so, for creating the necessary epistemic basis for trust at an individual level as much as among the public, who can have confidence that the AI systems have been subject to meaningful scrutiny. In this sense,

transparency was deemed to support trust among affected communities and stakeholders by providing the means to evaluate claims made about these systems with regard to bias and discrimination. AI systems reproducing inequalities and exacerbating injustices may, in turn, erode public trust not only in the AI-based tools themselves but also in stakeholders involved in building, deploying, and using such systems. Addressing and testing for such biases was deemed particularly relevant considering that “personal and collective experiences with discrimination or degradation along lines of race, class, gender, or other personal characteristics especially create reasons for suspicion if not outright distrust” [99]. However, as the experts agreed, such testing needs to go beyond the sole measure of excluding apparently discriminatory information. Deep learning models may predict a patient’s race from medical images such as chest and hand x-rays and mammograms [100] or identify patient self-reported race from redacted clinical notes [101] despite human experts being unable to do the same. Therefore, as indicated by the resource allocation case study, discrimination can occur even when models are apparently *race blind* through implicit proxy features, as is well understood and documented in algorithmic fairness research [120]. Moreover, discrimination through differential impact across subpopulations might occur precisely because models are insufficiently sensitive to inequitable distribution of social determinants of health along ethnic or racial lines [121].

Beyond these common themes, our cases highlight important contextual differences among trust in different kinds of medical AI. In particular, these relate to (1) framing factors, (2) previous knowledge of the targeted trusters demanding different levels of explicability and transparency, and (3) the different risk-benefit trade-offs in different environments. Sometimes, the risks in medicine are grave for individual patients, such as in an ICU setting. However, in the resource allocation setting, the risk-benefit trade-offs concern both individual-level consequences of allocating or denying a resource to an individual and group- or population-wide benefits or harms due to better or worse resource use.

Discussion

Principal Findings

Using conceptual framework analysis, our study provides the first consensus-based publication investigating trust in AI in the domain of health care. Commonalities and differences emerged from the examination of our case studies, particularly with respect to causes and effects of user trust. Across all cases, trust was found to enhance acceptance and adoption of AI technologies, and some factors such as a system’s accuracy were similarly deemed to be crucial for the trustworthiness of all discussed systems. Developers’ openness about a system’s internal workings, assumptions, and value judgments underlining the technical choices were also considered crucial across the case studies. Such transparency fosters alignment with the goals and values of health care professionals and patients; supports accountability mechanisms; and may help address biases and discrimination, which is essential for building trust among affected communities and stakeholders. However, our analysis

also highlights contextual differences among trust in various forms of AI in health care. These differences relate to a multitude of case-specific aspects, such as framing factors, previous knowledge of the trusters, and risk-benefit trade-offs in different environments, necessitating tailored approaches to establish trust and trustworthiness. This finding has implications for anyone aiming to foster trust in AI within the health care domain, from developers and health care professionals using AI systems to public health experts and regulators—trust-building measures should distinguish among different types of AI systems considering not only implied risk levels and legal responsibilities in case of errors but also the factors outlined in our conceptual framework: context, actors, discourse, and the mechanisms of trust building.

Implications

Our findings have implications for developers, patients and health care professionals, and regulators aiming to increase trust in medical AI systems. The developing companies should communicate openly about the algorithmic design and training of their AI systems and tailor their level of transparency to the communicative needs of their respective audiences [122]. Such needs will be very different depending on whether the AI is used by patients directly (assistive AI case), trained physicians (diagnostic and predictive AI cases), health economists and policy makers (resource allocation AI case), or potentially the interested public (public health AI case). Individual AI end users need to be educated on the technology’s limitations and carefully consider who bears the risk and responsibility of involving the AI—especially if it is not themselves.

At a regulatory level, distinguishing among different kinds of AI systems in medicine seems also crucial, regulating them according to the level of implied risk as currently proposed by the European Union AI Act—or possibly at a finer-grained level specific for health care [123]. Regulation is also needed to clarify who bears legal responsibility in the case of specific errors. While there is a clear responsibility by developers to avoid systematic errors as much as possible, for the foreseeable future, it will not be possible to design an AI system without any errors. Therefore, some responsibility also falls on health care providers who use the systems as part of their workflows to use them appropriately and with the right level of human oversight. This requires strengthening both the technical capabilities of health care practitioners to evaluate AI systems and the responsibilities placed on the developers of systems to openly document where a system may be expected to generate errors. Given the limits of both, proper certification of AI systems is furthermore crucial, allowing for trust in an overseeing institution to be extended to the system itself [124].

A final important point is that there should be a focus on institutions implementing these tools in a trustworthy manner, not on increasing trust. There are several reasons for this. First, trust could be increased through marketing and presentational gimmicks targeted at patients and clinicians, which would do nothing to increase trustworthiness [105]. Second, trust may be hard to achieve given prevailing and justified distrust in health service providers such as Optum and private health systems and marginalized communities’ distrust in health care systems due

to historic or entrenched disparities in outcomes or experiences of care [125]. In contrast, this may be less of a problem where there are higher levels of trust in health institutions that might implement AI systems [106]. Finally, the size and complexity of health systems complicate questions of where, why, and in whom the individual patient should place trust within these systems [87]. This creates challenges in assessing how far the design and implementation of an algorithmic system for resource allocation impacted the attitudes of trust in affected patients.

Limitations

While our findings aim to provide a starting point for fruitful further work on AI in health care, there are several limitations to our consensus process and its results. Of these limitations, 3 are related to the selection of experts for our consensus process. First, owing to the recruitment of participants working on trust in AI in health care, our work is biased toward a position that considers trust a useful and meaningful concept in the context of AI in the first place. While this position is supported by a substantive part of the literature [28,31,49,52], it should be noted that there are also outspoken critics of using the notion of trust with respect to AI in the first place [98,126,127]. Second, given the location of our workshop, our recruitment focused largely on experts from Europe, with only a small addition of participants from North America and Oceania. While this may reflect the central role that trust and trustworthiness have taken in regulatory debates across the European Union, it limits the generalizability of our findings to other contexts. A recent systematic review of empirical research on trust in AI also revealed a lack of diversity in the discussion surrounding this topic [26]. This review underscored the need for a broader range of perspectives to more comprehensively understand and address the complexities of trust in AI systems [26]. Third, to address the lack of agreement on trust in AI in health care among experts, our work focused on expert agreement, leaving out some other relevant stakeholders. As the backgrounds of our participants highlight, some stakeholders were actively involved, from developers to practicing clinicians, but others were not included, such as patients, nurses, or hospital administrators. Further work is needed to address this gap.

There were 2 additional limitations more conceptual in nature, and we thank an anonymous reviewer for flagging these. First, our framework does not distinguish between trust *before* and trust *during* use of an AI-based system, yet causal and framing factors of trust may evolve with the experience of using a technology over time. In fact, all our case studies focus more on the initial adoption of a technology and less on trust evolving during its use, reflecting the current early-stage integration of AI systems into health care. That being said, we do hold that our framework can adequately reflect the relevant factors of trust in different AI systems in health care—even if they may change over time. Second, our work did not focus on the phenomenon of distrust despite its undebatable influence on user acceptance and its irreconcilability with trust. We consciously omitted distrust as it is often considered to not be

a mere absence of trust but a more complex, richer phenomenon [19,23,95]. Therefore, we believe that a consensus on distrust deserves a paper of its own. However, it should be noted that distrust toward a specific technology, or potentially against AI in general, can of course play a crucial role in inhibiting trust building at a causal level.

Finally, when considering wider conceptual work on trust and the focus on warranted trust in this paper, we need to acknowledge a body of literature that understands trust to be motivated by emotions rather than calculated decisions [128]. The conceptual thinking in this paper leans more toward a cognitive approach to trust building. Thereby, the effect of affective states on trust might be somehow undervalued. However, these affective states can play an important role in trusting and accepting AI in medicine, although their contribution to trusting AI can vary greatly across different instances of human-AI interactions [129]. Therefore, it will be necessary to extend the conceptual thinking beyond cognitive approaches to encompass traits of affective trust.

In summary, our findings suggest that achieving trustworthy AI systems in health care requires a multifaceted approach bridging human-centered and technology-centered approaches. While regulatory precision and boundaries help provide the legal basis needed to develop trustworthy AI, and while technological design features are critical to successful AI development, communication and positive lived experiences with AI systems are at the forefront of user trust. From a user perspective, especially for those who are not AI aficionados, trustworthy technological features and legal compliance should be a sure thing, whereas communication and positive experiences point beyond the technical sphere of AI.

Conclusions

This paper examines 5 diverse AI systems in health care, revealing the complex landscape of trust in this field. These cases highlight how AI in health care is influenced by various environments, actors, and framing factors. While discussing trust in medical AI in a broad sense may serve as an initial overview, the specific nuances uncovered in these 5 cases demand a more detailed understanding to characterize trusting AI in health care.

Ultimately, however, focus should shift toward ensuring that institutions implement these tools in a *trustworthy* manner rather than merely aiming to indefinitely increase *trust*—only warranted trust is needed for valuable AI implementation. This approach recognizes the complexity of trust dynamics, the risk of superficial increase in trust without a substantial enhancement of trustworthiness, and the importance of trustworthiness in health care institutions. Despite all domain-specific variations, addressing these challenges and embracing transparency and accountability by design can help build a foundation of trust that will underpin the successful integration of AI into health care for the benefit of patients, physicians, and society at large.

Acknowledgments

This paper was produced within the framework of the workshop titled “To trust or not to trust: When should AI be allowed to make decisions?” funded by the Swiss National Science Foundation (SNSF) under grant IZSEZO_213480 (applicant: MI; coapplicants: A Facchini and FG) and with local support from the École Polytechnique Fédérale de Lausanne. At the time of the workshop, A Ferrario was affiliated with the Mobiliar Lab for Analytics at Federal Institute of Technology Zürich and gratefully acknowledges their support. FG acknowledges the support of the Digital Society Initiative, University of Zurich. Unrelated and outside of the work leading to this paper, FG acknowledges funding from Novartis International AG, Sanitas Krankenversicherung (Stiftung), the Swiss Academy of Engineering Sciences, and the World Health Organization. RC acknowledges the support of the Digitalization Initiative of the Zurich Higher Education Institutions fellowship program of Zurich University of Applied Sciences digital. KJ acknowledges funding from the Wilhelmina Onderzoeksfonds (grant wkz22040701). ER’s research is supported by a Medical Sciences Graduate School Studentship, issued by the Nuffield Department of Population Health, and the Baillie Gifford-Institute for Ethics in AI Scholarship, both associated with the University of Oxford. PT acknowledges funding from Warsaw University of Technology within the Excellence Initiative – Research University program (grant 1820/97/Z01/2023). A Facchini and AT acknowledge funding from the University of Applied Sciences and Arts of Southern Switzerland Best4EthicalAI research program. MI and GS also acknowledge funding from the ERA-NET NEURON network and the SNSF under grant 32NE30_199436 (HybridMinds). GS acknowledges generous support from Fondation Brocher, Hermance, to finish a first full draft during a research residence. The work reported in this paper was funded by the SNSF (grant 32NE30_199436).

Data Availability

The datasets generated during and/or analyzed during this study are available from the corresponding author on reasonable request. The asynchronous consensus-building process was tracked in repeatedly updated documents.

Authors' Contributions

GS, FG, AT, A Facchini, and MI conceived and planned the workshop and the consensus process. All authors provided input on the choice of case studies and the framework. GS coordinated the consensus process and compiled the individual authors' contributions for a first full draft of the manuscript. GS, AT, A Facchini, and MI drafted a first introduction, and GS, FG, AT, YSJA, JH, KJ, BK, EP, PT, JW, and A Facchini contributed text to the Methods section. KJ led the consensus for the first case study, JH led the consensus for the second case study, YSJA led the consensus for the third case study, GS led the consensus for the fourth case study, and JW led the consensus for the fifth case study. The discussion was jointly developed by GS, FG, AT, A Facchini, and MI. A Ferrario further contributed substantially to the development of the manuscript by providing extensive critical feedback during the review phase. All authors provided repeated critical feedback and read and approved the final manuscript.

Conflicts of Interest

None declared.

References

1. Topol EJ. High-performance medicine: the convergence of human and artificial intelligence. *Nat Med.* Jan 2019;25(1):44-56. [doi: [10.1038/s41591-018-0300-7](https://doi.org/10.1038/s41591-018-0300-7)] [Medline: [30617339](https://pubmed.ncbi.nlm.nih.gov/30617339/)]
2. Esteva A, Kuprel B, Novoa RA, Ko J, Swetter SM, Blau HM, et al. Dermatologist-level classification of skin cancer with deep neural networks. *Nature.* Feb 02, 2017;542(7639):115-118. [FREE Full text] [doi: [10.1038/nature21056](https://doi.org/10.1038/nature21056)] [Medline: [28117445](https://pubmed.ncbi.nlm.nih.gov/28117445/)]
3. Lång K, Josefsson V, Larsson AM, Larsson S, Högberg C, Sartor H, et al. Artificial intelligence-supported screen reading versus standard double reading in the Mammography Screening with Artificial Intelligence trial (MASAI): a clinical safety analysis of a randomised, controlled, non-inferiority, single-blinded, screening accuracy study. *Lancet Oncol.* Aug 2023;24(8):936-944. [doi: [10.1016/S1470-2045\(23\)00298-X](https://doi.org/10.1016/S1470-2045(23)00298-X)] [Medline: [37541274](https://pubmed.ncbi.nlm.nih.gov/37541274/)]
4. Komorowski M, Celi LA, Badawi O, Gordon AC, Faisal AA. The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care. *Nat Med.* Nov 22, 2018;24(11):1716-1720. [FREE Full text] [doi: [10.1038/s41591-018-0213-5](https://doi.org/10.1038/s41591-018-0213-5)] [Medline: [30349085](https://pubmed.ncbi.nlm.nih.gov/30349085/)]
5. Hyland SL, Faltys M, Hüser M, Lyu X, Gumbsch T, Esteban C, et al. Early prediction of circulatory failure in the intensive care unit using machine learning. *Nat Med.* Mar 09, 2020;26(3):364-373. [doi: [10.1038/s41591-020-0789-4](https://doi.org/10.1038/s41591-020-0789-4)] [Medline: [32152583](https://pubmed.ncbi.nlm.nih.gov/32152583/)]
6. Ienca M, Fabrice J, Elger B, Caon M, Scoccia Pappagallo A, Kressig RW, et al. Intelligent assistive technology for Alzheimer's disease and other dementias: a systematic review. *J Alzheimers Dis.* 2017;56(4):1301-1340. [doi: [10.3233/JAD-161037](https://doi.org/10.3233/JAD-161037)] [Medline: [28222516](https://pubmed.ncbi.nlm.nih.gov/28222516/)]
7. Valeriani D, Santoro F, Ienca M. The present and future of neural interfaces. *Front Neurobot.* Oct 11, 2022;16:953968. [FREE Full text] [doi: [10.3389/fnbot.2022.953968](https://doi.org/10.3389/fnbot.2022.953968)] [Medline: [36304780](https://pubmed.ncbi.nlm.nih.gov/36304780/)]

8. Zeng D, Cao Z, Neill DB. Artificial intelligence-enabled public health surveillance—from local detection to global epidemic monitoring and control. In: Xing L, Giger ML, Min JK, editors. *Artificial Intelligence in Medicine: Technical Basis and Clinical Applications*. New York, NY: Academic Press; 2021:2021-2053.
9. Thirunavukarasu AJ, Ting DS, Elangovan K, Gutierrez L, Tan TF, Ting DS. Large language models in medicine. *Nat Med*. Aug 17, 2023;29(8):1930-1940. [doi: [10.1038/s41591-023-02448-8](https://doi.org/10.1038/s41591-023-02448-8)] [Medline: [37460753](https://pubmed.ncbi.nlm.nih.gov/37460753/)]
10. Shani C, Zarecki J, Shahaf D. The lean data scientist: recent advances toward overcoming the data bottleneck. *Commun ACM*. Jan 20, 2023;66(2):92-102. [doi: [10.1145/3551635](https://doi.org/10.1145/3551635)]
11. van Velsen L, Ludden G, Grünloh C. The limitations of user-and human-centered design in an eHealth context and how to move beyond them. *J Med Internet Res*. Oct 05, 2022;24(10):e37341. [FREE Full text] [doi: [10.2196/37341](https://doi.org/10.2196/37341)] [Medline: [36197718](https://pubmed.ncbi.nlm.nih.gov/36197718/)]
12. Gama F, Tyskbo D, Nygren J, Barlow J, Reed J, Svedberg P. Implementation frameworks for artificial intelligence translation into health care practice: scoping review. *J Med Internet Res*. Jan 27, 2022;24(1):e32215. [FREE Full text] [doi: [10.2196/32215](https://doi.org/10.2196/32215)] [Medline: [35084349](https://pubmed.ncbi.nlm.nih.gov/35084349/)]
13. He J, Baxter SL, Xu J, Xu J, Zhou X, Zhang K. The practical implementation of artificial intelligence technologies in medicine. *Nat Med*. Jan 2019;25(1):30-36. [FREE Full text] [doi: [10.1038/s41591-018-0307-0](https://doi.org/10.1038/s41591-018-0307-0)] [Medline: [30617336](https://pubmed.ncbi.nlm.nih.gov/30617336/)]
14. Shaw J, Rudzicz F, Jamieson T, Goldfarb A. Artificial intelligence and the implementation challenge. *J Med Internet Res*. Jul 10, 2019;21(7):e13659. [FREE Full text] [doi: [10.2196/13659](https://doi.org/10.2196/13659)] [Medline: [31293245](https://pubmed.ncbi.nlm.nih.gov/31293245/)]
15. Chekroud AM, Hawrilenko M, Loho H, Bondar J, Gueorguieva R, Hasan A, et al. Illusory generalizability of clinical prediction models. *Science*. Jan 12, 2024;383(6679):164-167. [doi: [10.1126/science.adg8538](https://doi.org/10.1126/science.adg8538)] [Medline: [38207039](https://pubmed.ncbi.nlm.nih.gov/38207039/)]
16. Crossnohere NL, Elsaid M, Paskett J, Bose-Brill S, Bridges JFP. Guidelines for artificial intelligence in medicine: literature review and content analysis of frameworks. *J Med Internet Res*. Aug 25, 2022;24(8):e36823. [FREE Full text] [doi: [10.2196/36823](https://doi.org/10.2196/36823)] [Medline: [36006692](https://pubmed.ncbi.nlm.nih.gov/36006692/)]
17. Maddox TM, Rumsfeld JS, Payne PR. Questions for artificial intelligence in health care. *JAMA*. Jan 01, 2019;321(1):31-32. [doi: [10.1001/jama.2018.18932](https://doi.org/10.1001/jama.2018.18932)] [Medline: [30535130](https://pubmed.ncbi.nlm.nih.gov/30535130/)]
18. Tucci V, Saary J, Doyle TE. Factors influencing trust in medical artificial intelligence for healthcare professionals: a narrative review. *J Med Artif Intell*. Mar 2022;5:4. [doi: [10.21037/jmai-21-25](https://doi.org/10.21037/jmai-21-25)]
19. McLeod C. Stanford encyclopedia of philosophy. The Metaphysics Research Lab. URL: <https://plato.stanford.edu/archives/fall2021/entries/trust/> [accessed 2024-04-29]
20. McKnight DH, Chervany NL. What is trust? A conceptual analysis and an interdisciplinary model. In: *Proceedings of the 2000 Americas Conference on Information Systems*. 2000. Presented at: AMCIS '00; August 10-13, 2000:382; Long Beach, CA. URL: <https://aisel.aisnet.org/amcis2000/382>
21. Baier A. Tanner lectures on human values. Princeton University. 1991. URL: <https://uchv.princeton.edu/events/tanner-lectures-human-values> [accessed 2024-04-29]
22. Luhmann N. *Vertrauen : ein Mechanismus der Reduktion sozialer Komplexität*. Stuttgart, Germany. F. Enke; 1968.
23. Hawley K. Trust, distrust and commitment. *Noûs*. Oct 25, 2012;48(1):1-20. [FREE Full text] [doi: [10.1111/nous.12000](https://doi.org/10.1111/nous.12000)]
24. Castelfranchi C, Falcone R. *Trust Theory - A Socio-Cognitive and Computational Model*. Hoboken, NJ. John Wiley & Sons; 2010.
25. Jacovi A, Marasović A, Miller T, Goldberg Y. Formalizing trust in artificial intelligence: prerequisites, causes and goals of human trust in AI. In: *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*. 2021. Presented at: FAccT '21; March 3-10, 2021:624-635; Virtual Event. URL: <https://dl.acm.org/doi/10.1145/3442188.3445923> [doi: [10.1145/3442188.3445923](https://doi.org/10.1145/3442188.3445923)]
26. Benk M, Kerstan S, von Wangenheim F, Ferrario A. Twenty-four years of empirical research on trust in AI: a bibliometric review of trends, overlooked issues, and future directions. *AI Soc*. Oct 02, 2024;25. [FREE Full text] [doi: [10.1007/S00146-024-02059-Y](https://doi.org/10.1007/S00146-024-02059-Y)]
27. Jones C, Thornton J, Wyatt JC. Artificial intelligence and clinical decision support: clinicians' perspectives on trust, trustworthiness, and liability. *Med Law Rev*. Nov 27, 2023;31(4):501-520. [FREE Full text] [doi: [10.1093/medlaw/fwad013](https://doi.org/10.1093/medlaw/fwad013)] [Medline: [37218368](https://pubmed.ncbi.nlm.nih.gov/37218368/)]
28. Ferrario A, Loi M. How explainability contributes to trust in AI. In: *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*. 2022. Presented at: FAccT '22; June 21-24, 2022:1457-1466; Seoul, Republic of Korea. URL: <https://dl.acm.org/doi/10.1145/3531146.3533202> [doi: [10.1145/3531146.3533202](https://doi.org/10.1145/3531146.3533202)]
29. Starke G, Ienca M. Misplaced trust and distrust: how not to engage with medical artificial intelligence. *Camb Q Healthc Ethics*. Oct 20, 2022;33(3):1-10. [doi: [10.1017/S0963180122000445](https://doi.org/10.1017/S0963180122000445)] [Medline: [36263755](https://pubmed.ncbi.nlm.nih.gov/36263755/)]
30. Choung H, David P, Ross A. Trust in AI and its role in the acceptance of AI technologies. *Int J Hum Comput Interact*. Apr 20, 2022;39(9):1727-1739. [doi: [10.1080/10447318.2022.2050543](https://doi.org/10.1080/10447318.2022.2050543)]
31. Gille F, Jobin A, Ienca M. What we talk about when we talk about trust: theory of trust for AI in healthcare. *Intell Based Med*. Nov 2020;1-2:100001. [doi: [10.1016/j.ibmed.2020.100001](https://doi.org/10.1016/j.ibmed.2020.100001)]
32. Lee JD, See KA. Trust in automation: designing for appropriate reliance. *Hum Factors*. 2004;46(1):50-80. [doi: [10.1518/hfes.46.1.50.30392](https://doi.org/10.1518/hfes.46.1.50.30392)] [Medline: [15151155](https://pubmed.ncbi.nlm.nih.gov/15151155/)]

33. Gibson GR, Hutkins R, Sanders ME, Prescott SL, Reimer RA, Salminen SJ, et al. Expert consensus document: the International Scientific Association for Probiotics and Prebiotics (ISAPP) consensus statement on the definition and scope of prebiotics. *Nat Rev Gastroenterol Hepatol*. Aug 14, 2017;14(8):491-502. [FREE Full text] [doi: [10.1038/nrgastro.2017.75](https://doi.org/10.1038/nrgastro.2017.75)] [Medline: [28611480](https://pubmed.ncbi.nlm.nih.gov/28611480/)]
34. Giustina A, Chanson P, Kleinberg D, Bronstein MD, Clemmons DR, Klibanski A, et al. Acromegaly Consensus Group. Expert consensus document: a consensus on the medical treatment of acromegaly. *Nat Rev Endocrinol*. Apr 25, 2014;10(4):243-248. [doi: [10.1038/nrendo.2014.21](https://doi.org/10.1038/nrendo.2014.21)] [Medline: [24566817](https://pubmed.ncbi.nlm.nih.gov/24566817/)]
35. Ienca M, Fins JJ, Jox RJ, Jotterand F, Voienky S, Andorno R, et al. Towards a governance framework for brain data. *Neuroethics*. Jun 03, 2022;15(2):20. [doi: [10.1007/S12152-022-09498-8](https://doi.org/10.1007/S12152-022-09498-8)]
36. Ives J, Dunn M, Molewijk B, Schildmann J, Børøe K, Frith L, et al. Standards of practice in empirical bioethics research: towards a consensus. *BMC Med Ethics*. Jul 10, 2018;19(1):68. [FREE Full text] [doi: [10.1186/s12910-018-0304-3](https://doi.org/10.1186/s12910-018-0304-3)] [Medline: [29986689](https://pubmed.ncbi.nlm.nih.gov/29986689/)]
37. Gattrell WT, Logullo P, van Zuuren EJ, Price A, Hughes EL, Blazey P, et al. ACCORD (ACcurate CONsensus reporting document): a reporting guideline for consensus methods in biomedicine developed via a modified Delphi. *PLoS Med*. Jan 23, 2024;21(1):e1004326. [FREE Full text] [doi: [10.1371/journal.pmed.1004326](https://doi.org/10.1371/journal.pmed.1004326)] [Medline: [38261576](https://pubmed.ncbi.nlm.nih.gov/38261576/)]
38. La Brooy C, Pratt B, Kelaher M. What is the role of consensus statements in a risk society? *J Risk Res*. Jul 25, 2019;23(5):664-677. [doi: [10.1080/13669877.2019.1628094](https://doi.org/10.1080/13669877.2019.1628094)]
39. Tong A, Sainsbury P, Craig J. Consolidated criteria for reporting qualitative research (COREQ): a 32-item checklist for interviews and focus groups. *Int J Qual Health Care*. Dec 2007;19(6):349-357. [doi: [10.1093/intqhc/mzm042](https://doi.org/10.1093/intqhc/mzm042)] [Medline: [17872937](https://pubmed.ncbi.nlm.nih.gov/17872937/)]
40. Ancker JS, Benda NC, Reddy M, Unertl KM, Veinot T. Guidance for publishing qualitative research in informatics. *J Am Med Inform Assoc*. Nov 25, 2021;28(12):2743-2748. [FREE Full text] [doi: [10.1093/jamia/ocab195](https://doi.org/10.1093/jamia/ocab195)] [Medline: [34537840](https://pubmed.ncbi.nlm.nih.gov/34537840/)]
41. Jabareen Y. Building a conceptual framework: philosophy, definitions, and procedure. *Int J Qual Methods*. Dec 01, 2009;8(4):49-62. [doi: [10.1177/160940690900800406](https://doi.org/10.1177/160940690900800406)]
42. U.S. Department of HealthHuman Services FDA Center for Drug EvaluationResearch, U.S. Department of HealthHuman Services FDA Center for Biologics EvaluationResearch, U.S. Department of HealthHuman Services FDA Center for DevicesRadiological Health. Guidance for industry: patient-reported outcome measures: use in medical product development to support labeling claims: draft guidance. *Health Qual Life Outcomes*. Oct 11, 2006;4(1):79. [FREE Full text] [doi: [10.1186/1477-7525-4-79](https://doi.org/10.1186/1477-7525-4-79)] [Medline: [17034633](https://pubmed.ncbi.nlm.nih.gov/17034633/)]
43. OECD guidelines on measuring trust. Organisation for Economic Co-operation and Development. URL: https://www.oecd.org/en/publications/oecd-guidelines-on-measuring-trust_9789264278219-en.html [accessed 2024-04-29]
44. Gille F, Smith S, Mays N. Why public trust in health care systems matters and deserves greater research attention. *J Health Serv Res Policy*. Jan 17, 2015;20(1):62-64. [doi: [10.1177/1355819614543161](https://doi.org/10.1177/1355819614543161)] [Medline: [25038059](https://pubmed.ncbi.nlm.nih.gov/25038059/)]
45. Ferrario A, Loi M, Viganò E. In AI we trust incrementally: a multi-layer model of trust to analyze human-artificial intelligence interactions. *Philos Technol*. Oct 23, 2019;33(3):523-539. [doi: [10.1007/S13347-019-00378-3](https://doi.org/10.1007/S13347-019-00378-3)]
46. Ferrario A, Loi M, Viganò E. Trust does not need to be human: it is possible to trust medical AI. *J Med Ethics*. Nov 25, 2020;47(6):437-438. [FREE Full text] [doi: [10.1136/medethics-2020-106922](https://doi.org/10.1136/medethics-2020-106922)] [Medline: [33239471](https://pubmed.ncbi.nlm.nih.gov/33239471/)]
47. Gille F. What Is Public Trust in the Health System?: Insights into Health Data Use. Bristol, UK. Policy Press; 2023.
48. Durán JM, Formanek N. Grounds for trust: essential epistemic opacity and computational reliabilism. *Minds Mach*. Oct 29, 2018;28(4):645-666. [doi: [10.1007/S11023-018-9481-6](https://doi.org/10.1007/S11023-018-9481-6)]
49. Durán JM, Jongsma KR. Who is afraid of black box algorithms? On the epistemological and ethical basis of trust in medical AI. *J Med Ethics*. Mar 18, 2021;22. [doi: [10.1136/medethics-2020-106820](https://doi.org/10.1136/medethics-2020-106820)] [Medline: [33737318](https://pubmed.ncbi.nlm.nih.gov/33737318/)]
50. Loi M, Ferrario A, Viganò E. How much do you trust me? A logico-mathematical analysis of the concept of the intensity of trust. *Synthese*. May 23, 2023;201(6):186. [doi: [10.1007/S11229-023-04169-4](https://doi.org/10.1007/S11229-023-04169-4)]
51. Chong D, Druckman JN. Framing theory. *Annu Rev Polit Sci*. Jun 01, 2007;10(1):103-126. [doi: [10.1146/annurev.polisci.10.072805.103054](https://doi.org/10.1146/annurev.polisci.10.072805.103054)]
52. Starke G, van den Brule R, Elger BS, Haselager P. Intentional machines: a defence of trust in medical artificial intelligence. *Bioethics*. Feb 18, 2022;36(2):154-161. [doi: [10.1111/bioe.12891](https://doi.org/10.1111/bioe.12891)] [Medline: [34142373](https://pubmed.ncbi.nlm.nih.gov/34142373/)]
53. Hosny A, Parmar C, Quackenbush J, Schwartz LH, Aerts HJ. Artificial intelligence in radiology. *Nat Rev Cancer*. Aug 2018;18(8):500-510. [FREE Full text] [doi: [10.1038/s41568-018-0016-5](https://doi.org/10.1038/s41568-018-0016-5)] [Medline: [29777175](https://pubmed.ncbi.nlm.nih.gov/29777175/)]
54. Jiang F, Jiang Y, Zhi H, Dong Y, Li H, Ma S, et al. Artificial intelligence in healthcare: past, present and future. *Stroke Vasc Neurol*. Dec 2017;2(4):230-243. [FREE Full text] [doi: [10.1136/svn-2017-000101](https://doi.org/10.1136/svn-2017-000101)] [Medline: [29507784](https://pubmed.ncbi.nlm.nih.gov/29507784/)]
55. Pesapane F, Codari M, Sardanelli F. Artificial intelligence in medical imaging: threat or opportunity? Radiologists again at the forefront of innovation in medicine. *Eur Radiol Exp*. Oct 24, 2018;2(1):35. [FREE Full text] [doi: [10.1186/s41747-018-0061-6](https://doi.org/10.1186/s41747-018-0061-6)] [Medline: [30353365](https://pubmed.ncbi.nlm.nih.gov/30353365/)]
56. McDonald ES, McCarthy AM, Akhtar AL, Synnvestedt MB, Schnall M, Conant EF. Baseline screening mammography: performance of full-field digital mammography versus digital breast tomosynthesis. *AJR Am J Roentgenol*. Nov 2015;205(5):1143-1148. [doi: [10.2214/AJR.15.14406](https://doi.org/10.2214/AJR.15.14406)] [Medline: [26496565](https://pubmed.ncbi.nlm.nih.gov/26496565/)]

57. Tang A, Tam R, Cadrin-Chênevert A, Guest W, Chong J, Barfett J, et al. Canadian Association of Radiologists (CAR) Artificial Intelligence Working Group. Canadian association of radiologists white paper on artificial intelligence in radiology. *Can Assoc Radiol J*. May 2018;69(2):120-135. [FREE Full text] [doi: [10.1016/j.carj.2018.02.002](https://doi.org/10.1016/j.carj.2018.02.002)] [Medline: [29655580](#)]
58. Bejnordi BE, Veta M, Johannes van Diest P, van Ginneken B, Karssemeijer N, Litjens G, the CAMELYON16 Consortium, et al. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *JAMA*. Dec 12, 2017;318(22):2199-2210. [FREE Full text] [doi: [10.1001/jama.2017.14585](https://doi.org/10.1001/jama.2017.14585)] [Medline: [29234806](#)]
59. Ghafoorian M, Karssemeijer N, Heskes T, Bergkamp M, Wissink J, Obels J, et al. Deep multi-scale location-aware 3D convolutional neural networks for automated detection of lacunes of presumed vascular origin. *Neuroimage Clin*. 2017;14:391-399. [FREE Full text] [doi: [10.1016/j.nicl.2017.01.033](https://doi.org/10.1016/j.nicl.2017.01.033)] [Medline: [28271039](#)]
60. Oxipit awarded CE mark for first autonomous AI medical imaging application. Oxipit.ai. URL: <https://oxipit.ai/news/first-autonomous-ai-medical-imaging-application/> [accessed 2024-04-29]
61. Keski-Filppula T, Nikki M, Haapea M, Ramanauskas N, Tervonen O. Using artificial intelligence to detect chest X-rays with no significant findings in a primary health care setting in Oulu, Finland. *arXiv*. Preprint posted online May 17, 2022. [FREE Full text] [doi: [10.48550/arXiv.2205.08123](https://doi.org/10.48550/arXiv.2205.08123)]
62. Grote T, Berens P. On the ethics of algorithmic decision-making in healthcare. *J Med Ethics*. Mar 2020;46(3):205-211. [FREE Full text] [doi: [10.1136/medethics-2019-105586](https://doi.org/10.1136/medethics-2019-105586)] [Medline: [31748206](#)]
63. Krishnan R, Rajpurkar P, Topol EJ. Self-supervised learning in medicine and healthcare. *Nat Biomed Eng*. Dec 11, 2022;6(12):1346-1352. [doi: [10.1038/s41551-022-00914-1](https://doi.org/10.1038/s41551-022-00914-1)] [Medline: [35953649](#)]
64. Tizhoosh HR, Pantanowitz L. Artificial intelligence and digital pathology: challenges and opportunities. *J Pathol Inform*. 2018;9:38. [FREE Full text] [doi: [10.4103/jpi.jpi_53_18](https://doi.org/10.4103/jpi.jpi_53_18)] [Medline: [30607305](#)]
65. Pena GP, Andrade-Filho JS. How does a pathologist make a diagnosis? *Arch Pathol Lab Med*. Jan 2009;133(1):124-132. [FREE Full text] [doi: [10.5858/133.1.124](https://doi.org/10.5858/133.1.124)] [Medline: [19123724](#)]
66. Arbelaez Ossa L, Starke G, Lorenzini G, Vogt JE, Shaw DM, Elger BS. Re-focusing explainability in medicine. *Digit Health*. Feb 11, 2022;8:20552076221074488. [FREE Full text] [doi: [10.1177/20552076221074488](https://doi.org/10.1177/20552076221074488)] [Medline: [35173981](#)]
67. Starke G, Elger BS, De Clercq E. Machine learning and its impact on psychiatric nosology: findings from a qualitative study among German and Swiss experts. *PhiMiSci*. Apr 11, 2023;4:1-17. [doi: [10.33735/philimisci.2023.9435](https://doi.org/10.33735/philimisci.2023.9435)]
68. Walker MJ, Rogers WA. Diagnosis, narrative identity, and asymptomatic disease. *Theor Med Bioeth*. Aug 5, 2017;38(4):307-321. [doi: [10.1007/s11017-017-9412-1](https://doi.org/10.1007/s11017-017-9412-1)] [Medline: [28681328](#)]
69. Drogts J, Milota M, Vos S, Bredenoord A, Jongsma K. Integrating artificial intelligence in pathology: a qualitative interview study of users' experiences and expectations. *Mod Pathol*. Nov 2022;35(11):1540-1550. [FREE Full text] [doi: [10.1038/s41379-022-01123-6](https://doi.org/10.1038/s41379-022-01123-6)] [Medline: [35927490](#)]
70. Starke G, Schmidt B, De Clercq E, Elger BS. Explainability as fig leaf? An exploration of experts' ethical expectations towards machine learning in psychiatry. *AI Ethics*. Jun 07, 2022;3(1):303-314. [doi: [10.1007/s43681-022-00177-1](https://doi.org/10.1007/s43681-022-00177-1)]
71. Yang L, Ene IC, Arabi Belaghi R, Koff D, Stein N, Santaguida P. Stakeholders' perspectives on the future of artificial intelligence in radiology: a scoping review. *Eur Radiol*. Mar 21, 2022;32(3):1477-1495. [doi: [10.1007/s00330-021-08214-z](https://doi.org/10.1007/s00330-021-08214-z)] [Medline: [34545445](#)]
72. Aquino YS, Rogers WA, Braunack-Mayer A, Frazer H, Win KT, Houssami N, et al. Utopia versus dystopia: professional perspectives on the impact of healthcare artificial intelligence on clinical roles and skills. *Int J Med Inform*. Jan 2023;169:104903. [FREE Full text] [doi: [10.1016/j.ijmedinf.2022.104903](https://doi.org/10.1016/j.ijmedinf.2022.104903)] [Medline: [36343512](#)]
73. Patel D, Kher V, Desai B, Lei X, Cen S, Nanda N, et al. Machine learning based predictors for COVID-19 disease severity. *Sci Rep*. Feb 25, 2021;11(1):4673. [FREE Full text] [doi: [10.1038/s41598-021-83967-7](https://doi.org/10.1038/s41598-021-83967-7)] [Medline: [33633145](#)]
74. Jauk S, Kramer D, Avian A, Berghold A, Leodolter W, Schulz S. Correction to: technology acceptance of a machine learning algorithm predicting delirium in a clinical setting: a mixed-methods study. *J Med Syst*. Mar 10, 2021;45(4):52. [FREE Full text] [doi: [10.1007/s10916-021-01728-5](https://doi.org/10.1007/s10916-021-01728-5)] [Medline: [33740133](#)]
75. van Royen FS, Moons KG, Geersing GJ, van Smeden M. Developing, validating, updating and judging the impact of prognostic models for respiratory diseases. *Eur Respir J*. Jun 21, 2022;60(3):2200250. [doi: [10.1183/13993003.00250-2022](https://doi.org/10.1183/13993003.00250-2022)]
76. Mikhael PG, Wohlwend J, Yala A, Karstens L, Xiang J, Takigami AK, et al. Sybil: a validated deep learning model to predict future lung cancer risk from a single low-dose chest computed tomography. *J Clin Oncol*. Apr 20, 2023;41(12):2191-2200. [FREE Full text] [doi: [10.1200/JCO.22.01345](https://doi.org/10.1200/JCO.22.01345)] [Medline: [36634294](#)]
77. Marinovich ML, Wylie E, Lotter W, Pearce A, Carter SM, Lund H, et al. Artificial intelligence (AI) to enhance breast cancer screening: protocol for population-based cohort study of cancer detection. *BMJ Open*. Jan 03, 2022;12(1):e054005. [FREE Full text] [doi: [10.1136/bmjopen-2021-054005](https://doi.org/10.1136/bmjopen-2021-054005)] [Medline: [34980622](#)]
78. King OC, Mertens M. Self-fulfilling prophecy in practical and automated prediction. *Ethic Theory Moral Prac*. Jan 16, 2023;26(1):127-152. [doi: [10.1007/S10677-022-10359-9](https://doi.org/10.1007/S10677-022-10359-9)]
79. Steegen S, Tuerlinckx F, Gelman A, Vanpaemel W. Increasing transparency through a multiverse analysis. *Perspect Psychol Sci*. Sep 29, 2016;11(5):702-712. [doi: [10.1177/1745691616658637](https://doi.org/10.1177/1745691616658637)] [Medline: [27694465](#)]
80. Marx CT, Calmon FD, Ustun B. Predictive multiplicity in classification. In: *Proceedings of the 37 th International Conference on Machine Learning*. 2020. Presented at: PMLR '20; July 12-18, 2020:1-10; Vienna, Austria. URL: <https://proceedings.mlr.press/v119/marx20a.html>

81. Kulynych B, Hsu H, Troncoso C, Calmon FP. Arbitrary decisions are a hidden cost of differentially private training. In: Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency. 2023. Presented at: FAccT '23; June 12-15, 2023:1609-1623; Chicago, IL. URL: <https://dl.acm.org/doi/10.1145/3593013.3594103> [doi: [10.1145/3593013.3594103](https://doi.org/10.1145/3593013.3594103)]
82. Kim A, Chung KC, Keir C, Patrick DL. Patient-reported outcomes associated with cancer screening: a systematic review. *BMC Cancer*. Mar 01, 2022;22(1):223. [FREE Full text] [doi: [10.1186/s12885-022-09261-5](https://doi.org/10.1186/s12885-022-09261-5)] [Medline: [35232405](https://pubmed.ncbi.nlm.nih.gov/35232405/)]
83. Dickinson JA, Pimlott N, Grad R, Singh H, Szafran O, Wilson BJ, et al. Screening: when things go wrong. *Can Fam Physician*. Jul 2018;64(7):502-508. [FREE Full text] [Medline: [30002025](https://pubmed.ncbi.nlm.nih.gov/30002025/)]
84. Peng Y, Liu E, Peng S, Chen Q, Li D, Lian D. Using artificial intelligence technology to fight COVID-19: a review. *Artif Intell Rev*. Jan 03, 2022;55(6):4941-4977. [FREE Full text] [doi: [10.1007/s10462-021-10106-z](https://doi.org/10.1007/s10462-021-10106-z)] [Medline: [35002010](https://pubmed.ncbi.nlm.nih.gov/35002010/)]
85. MacIntyre CR, Lim S, Quigley A. Preventing the next pandemic: use of artificial intelligence for epidemic monitoring and alerts. *Cell Rep Med*. Dec 20, 2022;3(12):100867. [FREE Full text] [doi: [10.1016/j.xcrm.2022.100867](https://doi.org/10.1016/j.xcrm.2022.100867)] [Medline: [36543103](https://pubmed.ncbi.nlm.nih.gov/36543103/)]
86. Bhargavi BS, Moa A. Global outbreaks of zika infection by epidemic observatory (EpiWATCH), 2016-2019. *Glob. Biosecurity*. Oct 21, 2020;2:55. [doi: [10.31646/gbio.83](https://doi.org/10.31646/gbio.83)]
87. Lesmanawati DAS, Adam DC, Hooshmand E, Moa A, Kunasekaran MP, MacIntyre CR. The global epidemiology of hepatitis a outbreaks 2016-2018 and the utility of EpiWATCH as a rapid epidemic intelligence service. *Glob Biosecurity*. Mar 23, 2021;3(1):25. [doi: [10.31646/gbio.100](https://doi.org/10.31646/gbio.100)]
88. Kuhner D, Fiederer LD, Aldinger J, Burget F, Völker M, Schirrmeister RT, et al. A service assistant combining autonomous robotics, flexible goal formulation, and deep-learning-based brain-computer interfacing. *Robot Auton Syst*. Jun 2019;116:98-113. [doi: [10.1016/j.robot.2019.02.015](https://doi.org/10.1016/j.robot.2019.02.015)]
89. ISO 8549-3:2020: prosthetics and orthotics — vocabulary: part 3: terms relating to orthoses. International Organization for Standardization. 2022. URL: <https://www.iso.org/standard/79497.html> [accessed 2024-04-29]
90. Baniqued PD, Stanyer EC, Awais M, Alazmani A, Jackson AE, Mon-Williams MA, et al. Brain-computer interface robotics for hand rehabilitation after stroke: a systematic review. *J Neuroeng Rehabil*. Jan 23, 2021;18(1):15. [FREE Full text] [doi: [10.1186/s12984-021-00820-8](https://doi.org/10.1186/s12984-021-00820-8)] [Medline: [33485365](https://pubmed.ncbi.nlm.nih.gov/33485365/)]
91. Kellmeyer P, Mueller O, Feingold-Polak R, Levy-Tzedek S. Social robots in rehabilitation: a question of trust. *Sci Robot*. Aug 15, 2018;3(21):eaat1587. [doi: [10.1126/scirobotics.aat1587](https://doi.org/10.1126/scirobotics.aat1587)] [Medline: [33141717](https://pubmed.ncbi.nlm.nih.gov/33141717/)]
92. Carmena JM. Brain versus machine control. *PLoS Biol*. Dec 14, 2004;2(12):e430. [doi: [10.1371/journal.pbio.0020430](https://doi.org/10.1371/journal.pbio.0020430)]
93. Obermeyer Z, Powers B, Vogeli C, Mullainathan S. Dissecting racial bias in an algorithm used to manage the health of populations. *Science*. Oct 25, 2019;366(6464):447-453. [FREE Full text] [doi: [10.1126/science.aax2342](https://doi.org/10.1126/science.aax2342)] [Medline: [31649194](https://pubmed.ncbi.nlm.nih.gov/31649194/)]
94. The Lancet Digital Health. There is no such thing as race in health-care algorithms. *Lancet Digit Health*. Dec 2019;1(8):e375. [FREE Full text] [doi: [10.1016/S2589-7500\(19\)30201-8](https://doi.org/10.1016/S2589-7500(19)30201-8)] [Medline: [33323212](https://pubmed.ncbi.nlm.nih.gov/33323212/)]
95. Jones K. Trust as an Affective Attitude. *Ethics*. Oct 1996;107(1):4-25. [doi: [10.1086/233694](https://doi.org/10.1086/233694)]
96. Sheehan M, Friesen P, Balmer A, Cheeks C, Davidson S, Devereux J, et al. Trust, trustworthiness and sharing patient data for research. *J Med Ethics*. May 18, 2020;47(12):e26. [doi: [10.1136/medethics-2019-106048](https://doi.org/10.1136/medethics-2019-106048)] [Medline: [32424061](https://pubmed.ncbi.nlm.nih.gov/32424061/)]
97. McGeer V, Pettit P. The empowering theory of trust. In: Faulkner P, Simpson T, editors. *The Philosophy of Trust*. Oxford, UK: Oxford Academic Press; 2017:14-34.
98. Hatherley JJ. Limits of trust in medical AI. *J Med Ethics*. Jul 2020;46(7):478-481. [doi: [10.1136/medethics-2019-105935](https://doi.org/10.1136/medethics-2019-105935)] [Medline: [32220870](https://pubmed.ncbi.nlm.nih.gov/32220870/)]
99. Wadden JJ. What kind of artificial intelligence should we want for use in healthcare decision-making applications? *Can J Bioeth*. May 27, 2021;4(1):94-100. [doi: [10.7202/1077636ar](https://doi.org/10.7202/1077636ar)]
100. Ratti E, Graves M. Explainable machine learning practices: opening another black box for reliable medical AI. *AI Ethics*. Feb 15, 2022;2(4):801-814. [doi: [10.1007/S43681-022-00141-z](https://doi.org/10.1007/S43681-022-00141-z)]
101. McLennan S, Fiske A, Tigard D, Müller R, Haddadin S, Buyx A. Embedded ethics: a proposal for integrating ethics into the development of medical AI. *BMC Med Ethics*. Jan 26, 2022;23(1):6. [FREE Full text] [doi: [10.1186/s12910-022-00746-3](https://doi.org/10.1186/s12910-022-00746-3)] [Medline: [35081955](https://pubmed.ncbi.nlm.nih.gov/35081955/)]
102. Algorithmic impact assessment: a case study in healthcare. Ada Lovelace Institute. URL: <https://www.adalovelaceinstitute.org/report/algorithmic-impact-assessment-case-study-healthcare/> [accessed 2024-04-29]
103. Markus AF, Kors JA, Rijnbeek PR. The role of explainability in creating trustworthy artificial intelligence for health care: a comprehensive survey of the terminology, design choices, and evaluation strategies. *J Biomed Inform*. Jan 2021;113:103655. [FREE Full text] [doi: [10.1016/j.jbi.2020.103655](https://doi.org/10.1016/j.jbi.2020.103655)] [Medline: [33309898](https://pubmed.ncbi.nlm.nih.gov/33309898/)]
104. Gerke S. Health AI for good rather than evil? The need for a new regulatory framework for AI-based medical devices. *Yale J Health Policy Law Ethics*. 2012;20(2):432. [FREE Full text]
105. Watson DS, Krutzinna J, Bruce IN, Griffiths CE, McInnes IB, Barnes MR, et al. Clinical applications of machine learning algorithms: beyond the black box. *BMJ*. Mar 12, 2019;364:l886. [FREE Full text] [doi: [10.1136/bmj.l886](https://doi.org/10.1136/bmj.l886)] [Medline: [30862612](https://pubmed.ncbi.nlm.nih.gov/30862612/)]
106. Möllering G. The nature of trust: from Georg Simmel to a theory of expectation, interpretation and suspension. *Sociology*. 2001;35(2):403-420. [doi: [10.1017/S0038038501000190](https://doi.org/10.1017/S0038038501000190)]

107. Simmel G. Soziologie: Untersuchungen über die Formen der Vergesellschaftung. Berlin, Germany. Duncker und Humblot; 1908.
108. Nickel PJ. Trust in medical artificial intelligence: a discretionary account. *Ethics Inf Technol*. Jan 24, 2022;24(1):e56. [doi: [10.1007/S10676-022-09630-5](https://doi.org/10.1007/S10676-022-09630-5)]
109. Ethics guidelines for trustworthy AI. European Commission: Directorate-General for Communications Networks, Content and Technology. URL: <https://op.europa.eu/en/publication-detail/-/publication/d3988569-0434-11ea-8c1f-01aa75ed71a1/language-en> [accessed 2024-04-29]
110. Braun M, Bleher H, Hummel P. A leap of faith: is there a formula for "trustworthy" AI? *Hastings Cent Rep*. May 19, 2021;51(3):17-22. [doi: [10.1002/hast.1207](https://doi.org/10.1002/hast.1207)] [Medline: [33606288](https://pubmed.ncbi.nlm.nih.gov/33606288/)]
111. O'Neill O. Linking trust to trustworthiness. *Int J Philos Stud*. Apr 25, 2018;26(2):293-300. [doi: [10.1080/09672559.2018.1454637](https://doi.org/10.1080/09672559.2018.1454637)]
112. Amann J, Blasimme A, Vayena E, Frey D, Madai VI, Precise4Q consortium. Explainability for artificial intelligence in healthcare: a multidisciplinary perspective. *BMC Med Inform Decis Mak*. Nov 30, 2020;20(1):310. [FREE Full text] [doi: [10.1186/s12911-020-01332-6](https://doi.org/10.1186/s12911-020-01332-6)] [Medline: [33256715](https://pubmed.ncbi.nlm.nih.gov/33256715/)]
113. Mincu D, Roy S. Developing robust benchmarks for driving forward AI innovation in healthcare. *Nat Mach Intell*. Nov 15, 2022;4(11):916-921. [doi: [10.1038/S42256-022-00559-4](https://doi.org/10.1038/S42256-022-00559-4)]
114. Strickland E. IBM Watson, heal thyself: how IBM overpromised and underdelivered on AI health care. *IEEE Spectr*. Apr 2019;56(4):24-31. [doi: [10.1109/mspec.2019.8678513](https://doi.org/10.1109/mspec.2019.8678513)]
115. Parikh RB, Teeple S, Navathe AS. Addressing bias in artificial intelligence in health care. *JAMA*. Dec 24, 2019;322(24):2377-2378. [FREE Full text] [doi: [10.1001/jama.2019.18058](https://doi.org/10.1001/jama.2019.18058)] [Medline: [31755905](https://pubmed.ncbi.nlm.nih.gov/31755905/)]
116. Vayena E, Blasimme A, Cohen IG. Machine learning in medicine: addressing ethical challenges. *PLoS Med*. Nov 2018;15(11):e1002689. [FREE Full text] [doi: [10.1371/journal.pmed.1002689](https://doi.org/10.1371/journal.pmed.1002689)] [Medline: [30399149](https://pubmed.ncbi.nlm.nih.gov/30399149/)]
117. Char DS, Abramoff MD, Feudtner C. Identifying ethical considerations for machine learning healthcare applications. *Am J Bioeth*. Nov 2020;20(11):7-17. [FREE Full text] [doi: [10.1080/15265161.2020.1819469](https://doi.org/10.1080/15265161.2020.1819469)] [Medline: [33103967](https://pubmed.ncbi.nlm.nih.gov/33103967/)]
118. Powell J. Trust me, I'm a chatbot: how artificial intelligence in health care fails the Turing test. *J Med Internet Res*. Oct 28, 2019;21(10):e16222. [FREE Full text] [doi: [10.2196/16222](https://doi.org/10.2196/16222)] [Medline: [31661083](https://pubmed.ncbi.nlm.nih.gov/31661083/)]
119. Rudin C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat Mach Intell*. May 13, 2019;1(5):206-215. [FREE Full text] [doi: [10.1038/s42256-019-0048-x](https://doi.org/10.1038/s42256-019-0048-x)] [Medline: [35603010](https://pubmed.ncbi.nlm.nih.gov/35603010/)]
120. Tschantz MC. What is proxy discrimination? In: *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*. 2022. Presented at: FAccT '22; June 21-24, 2022:1993; Seoul, Republic of Korea. URL: <https://dl.acm.org/doi/fullHtml/10.1145/3531146.3533242> [doi: [10.1145/3531146.3533242](https://doi.org/10.1145/3531146.3533242)]
121. Suleman M, Qureshi Z. Should medicine be colour blind? *J Med Ethics*. Nov 23, 2023;49(11):725-726. [doi: [10.1136/jme-2023-109634](https://doi.org/10.1136/jme-2023-109634)] [Medline: [37871944](https://pubmed.ncbi.nlm.nih.gov/37871944/)]
122. Starke G. The emperor's new clothes? Transparency and trust in machine learning for clinical neuroscience. In: Friedrich O, Wolkenstein A, Bublitz C, Jox RF, Racine E, editors. *Clinical Neurotechnology meets Artificial Intelligence: Philosophical, Ethical, Legal and Social Implications*. Cham, Switzerland. Springer; 2021:2021-2096.
123. Reddy S. Navigating the AI revolution: the case for precise regulation in health care. *J Med Internet Res*. Sep 11, 2023;25:e49989. [FREE Full text] [doi: [10.2196/49989](https://doi.org/10.2196/49989)] [Medline: [37695650](https://pubmed.ncbi.nlm.nih.gov/37695650/)]
124. Scharowski N, Benk M, Kühne SJ, Wettstein L, Brühlmann F. Certification labels for trustworthy ai: Insights from an empirical mixed-method study. In: *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*. 2023. Presented at: FAccT '23; June 12-15, 2023:248-260; Chicago, IL. URL: <https://dl.acm.org/doi/10.1145/3593013.3593994> [doi: [10.1145/3593013.3593994](https://doi.org/10.1145/3593013.3593994)]
125. Lee MK, Rich K. Who is included in human perceptions of AI?: Trust and perceived fairness around healthcare AI and cultural mistrust. In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 2021. Presented at: CHI '21; May 8-13, 2021:1-14; Yokohama, Japan. URL: <https://dl.acm.org/doi/10.1145/3411764.3445570> [doi: [10.1145/3411764.3445570](https://doi.org/10.1145/3411764.3445570)]
126. Ryan M. In AI we trust: ethics, artificial intelligence, and reliability. *Sci Eng Ethics*. Oct 10, 2020;26(5):2749-2767. [FREE Full text] [doi: [10.1007/s11948-020-00228-y](https://doi.org/10.1007/s11948-020-00228-y)] [Medline: [32524425](https://pubmed.ncbi.nlm.nih.gov/32524425/)]
127. Metzinger T. Ethics washing made in Europe. *Der Tagesspiegel*. URL: <https://www.tagesspiegel.de/politik/ethics-washing-made-in-europe-5937028.html> [accessed 2024-04-29]
128. PytlikZillig LM, Kimbrough CD. Consensus on conceptualizations and definitions of trust: are we there yet? In: Shockley E, Neal TM, PytlikZillig LM, Bornstein BH, editors. *Interdisciplinary Perspectives on Trust: Towards Theoretical and Methodological Integration*. Cham, Switzerland. Springer; 2016:17-47.
129. Kyung N, Kwon HE. Rationally trust, but emotionally? The roles of cognitive and affective trust in laypeople's acceptance of AI for preventive care operations. *Prod Oper Manag*. Jul 31, 2022;13785. [FREE Full text] [doi: [10.1111/poms.13785](https://doi.org/10.1111/poms.13785)]

Abbreviations

AI: artificial intelligence

BCI: brain-computer interface

ICU: intensive care unit

Edited by N Cahill, T Leung; submitted 12.01.24; peer-reviewed by H Burkhardt, P Nickel, L Weinert; comments to author 29.04.24; revised version received 31.07.24; accepted 28.11.24; published 19.02.25

Please cite as:

Starke G, Gille F, Termine A, Aquino YSJ, Chavarriaga R, Ferrario A, Hastings J, Jongsma K, Kellmeyer P, Kulynych B, Postan E, Racine E, Sahin D, Tomaszewska P, Vold K, Webb J, Facchini A, Ienca M

Finding Consensus on Trust in AI in Health Care: Recommendations From a Panel of International Experts

J Med Internet Res 2025;27:e56306

URL: <https://www.jmir.org/2025/1/e56306>

doi: [10.2196/56306](https://doi.org/10.2196/56306)

PMID: [39969962](https://pubmed.ncbi.nlm.nih.gov/39969962/)

©Georg Starke, Felix Gille, Alberto Termine, Yves Saint James Aquino, Ricardo Chavarriaga, Andrea Ferrario, Janna Hastings, Karin Jongsma, Philipp Kellmeyer, Bogdan Kulynych, Emily Postan, Elise Racine, Derya Sahin, Paulina Tomaszewska, Karina Vold, Jamie Webb, Alessandro Facchini, Marcello Ienca. Originally published in the Journal of Medical Internet Research (<https://www.jmir.org>), 19.02.2025. This is an open-access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work, first published in the Journal of Medical Internet Research (ISSN 1438-8871), is properly cited. The complete bibliographic information, a link to the original publication on <https://www.jmir.org/>, as well as this copyright and license information must be included.

ACCEPTED MANUSCRIPT • OPEN ACCESS

The NERVE-ML (Neural Engineering Reproducibility and Validity Essentials for Machine Learning) Checklist: Ensuring Machine Learning Advances Neural Engineering

To cite this article before publication: David Edwin Carlson *et al* 2025 *J. Neural Eng.* in press <https://doi.org/10.1088/1741-2552/adbfbf>

Manuscript version: Accepted Manuscript

Accepted Manuscript is "the version of the article accepted for publication including all changes made as a result of the peer review process, and which may also include the addition to the article by IOP Publishing of a header, an article ID, a cover sheet and/or an 'Accepted Manuscript' watermark, but excluding any other editing, typesetting or other changes made by IOP Publishing and/or its licensors"

This Accepted Manuscript is © 2025 The Author(s). Published by IOP Publishing Ltd.



As the Version of Record of this article is going to be / has been published on a gold open access basis under a CC BY 4.0 licence, this Accepted Manuscript is available for reuse under a CC BY 4.0 licence immediately.

Everyone is permitted to use all or part of the original content in this article, provided that they adhere to all the terms of the licence <https://creativecommons.org/licenses/by/4.0>

Although reasonable endeavours have been taken to obtain all necessary permissions from third parties to include their copyrighted content within this article, their full citation and copyright line may not be present in this Accepted Manuscript version. Before using any content from this article, please refer to the Version of Record on IOPscience once published for full citation and copyright details, as permissions may be required. All third party content is fully copyright protected and is not published on a gold open access basis under a CC BY licence, unless that is specifically stated in the figure caption in the Version of Record.

View the [article online](#) for updates and enhancements.

The NERVE-ML (Neural Engineering Reproducibility and Validity Essentials for Machine Learning) Checklist: Ensuring Machine Learning Advances Neural Engineering

David E. Carlson^{1,2}, Ricardo Chavarriaga³, Yiling Liu⁴, Fabien Lotte^{5,6}, Bao-Liang Lu^{7,8}

Authors are listed alphabetically

¹Department of Biostatistics and Bioinformatics, Duke University School of Medicine, Durham, NC, USA

²Department of Computer Science, Department of Civil and Environmental Engineering, Duke University, Durham, NC, USA

³Centre for Artificial Intelligence, School of Engineering, Zurich University of Applied Sciences ZHAW, Winterthur, Switzerland

⁴Program in Computational Biology and Bioinformatics, Duke University School of Medicine, Durham, NC, USA

⁵Inria Center at the University of Bordeaux, Talence, 33405, France

⁶LaBRI (CNRS / Univ. Bordeaux / Bordeaux INP), Talence, 33405, France

⁷Center for Brain-Like Computing and Machine Intelligence, Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China.

⁸Ruijin-Mihoyo Laboratory, Clinical Neuroscience Center, Ruijin Hospital, Shanghai Jiao Tong University School of Medicine, Shanghai 200020, China

Email: david.carlson@duke.edu

Abstract:

Objective. Machine learning's ability to capture intricate patterns makes it vital in neural engineering research. With its increasing use, ensuring the validity and reproducibility of machine learning methods is critical. Unfortunately, this has not always been the case in practice, as there have been recent retractions across various scientific fields due to the misuse of machine learning methods and validation procedures. To address these concerns, we propose the first version of the Neural Engineering Reproducibility and Validity Essentials for Machine Learning (NERVE-ML) checklist, a framework designed to promote the transparent, reproducible, and valid application of machine learning in neural engineering. **Approach.** We highlight some of the unique challenges of model validation in neural engineering, including the difficulties from limited subject numbers, repeated or non-independent samples, and high subject heterogeneity. Through detailed case studies, we demonstrate how different validation approaches can lead to divergent scientific conclusions, highlighting the importance of selecting appropriate procedures guided by the NERVE-ML checklist. Effectively addressing these challenges and properly scoping scientific conclusions will ensure that machine learning contributes to, rather than hinders, progress in neural engineering. **Main Results.** Our case studies demonstrate that improper validation approaches can result in flawed studies or overclaimed scientific conclusions, complicating the scientific discourse. The NERVE-ML checklist effectively addresses these concerns by providing guidelines to ensure that machine learning approaches in neural engineering are reproducible and lead to valid scientific conclusions. **Significance.** By effectively addressing these challenges and properly scoping scientific conclusions guided by the NERVE-ML checklist, we aim to help pave the way for a future where machine learning reliably enhances the quality and impact of neural engineering research.

1. Introduction:

The use of Artificial Intelligence (AI) through machine learning is now a critical tool in neural engineering, mainly because of its ability to capture complex patterns in neural data more flexibly than traditional data analysis methods. Its popularity and usage continue to grow each year, as measured by the number of articles mentioning machine learning^a. As machine learning becomes increasingly integrated into neural engineering, it is crucial to develop and enforce high standards for research validity and reproducibility in machine learning models to ensure that these techniques improve and accelerate the field.

Incorrect application of machine learning can lead to wrong conclusions, setting back scientific progress and potentially leading to incorrect follow-up research or worse, including failed attempts to translate these findings into applications. While the computational methods themselves have changed, this fact is a natural extension of the long-standing challenges of incorrect statistical conclusions¹ and the prevalence of unreproducible results². This problem is highlighted by the number of papers retracted due to errors in the machine learning validation procedure, from data handling mistakes to severe biases in data, algorithms or evaluation procedures. Additionally, many papers with incorrect conclusions have not been withdrawn, and such papers cloud the scientific record³.

For the sake of rigorous science, the use of machine learning in neural engineering research must be transparent, reproducible, and properly scoped. The scientific conclusions drawn from these models need to be valid. Limitations need to be understood and disclosed. Upholding these principles will ensure quality research and build trust in machine learning within the broad scientific community. This parallels efforts more broadly in society, where trustworthy artificial intelligence is needed to promote positive societal impact^{4,5}, as evidenced by the European efforts of the General Data Protection Regulation and the AI Act and the United States' efforts in the Blueprint for an AI Bill of Rights.

There have been numerous studies and reviews on using machine learning in neuroscience and neural engineering^{6,7}. We complement this body of work by providing guidelines that focus on how to validate the use of machine learning to ensure reproducibility and validity in neural engineering research. Similar challenges are currently being addressed in a wide variety of disciplines. This effort aligns with initiatives in the machine learning community to create guidelines to address these issues⁸, which have been adopted by many machine learning venues, including Neural Information Processing Systems and the International Conference on Machine Learning, among others. Likewise, there have been significant efforts to create checklists to encourage consideration of reproducibility of results in predictive models in medicine^{9,10}. There are several ongoing efforts to adapt such checklists to the particular use of AI and machine learning models¹¹. However, neural engineering presents unique challenges, such as working with a small number of subjects and data, variability in devices, possible lack of statistical independence between samples and differences between individuals. These unique

^a For example, at the Journal of Neural Engineering, 60% of all articles retrieved from a search on "machine learning" had been published in the last 5 years (accessed 9/27/24).

challenges require that we tailor the standard machine learning guidelines to neural engineering, as proposed here.

Our goal is to ensure that machine learning positively impacts neural engineering by *providing a robust framework for study scoping and interpretation*. We first provide details on machine learning practices and how inappropriately handling some of these challenges can cause incorrect interpretations through case studies and simulations. We then suggest best practices and introduce the Neural Engineering Reproducibility and Validity Essentials for Machine Learning (NERVE-ML) checklist. This checklist aims to guide researchers in ensuring their machine learning research in neural engineering is reproducible and the conclusions are valid. While it does not cover every situation, we believe that this is a step towards ensuring reliable machine learning.

Our effort complements other ongoing and recent initiatives within the field of neural engineering. For instance, the Mother of All BCI Benchmarks (MOABB)^{12–14} provides a standardized framework for benchmarking machine learning algorithms specifically within brain-computer interface (BCI) research, emphasizing reproducibility and cross-dataset comparability. Additionally, Roy et al.² provide a systematic review of deep learning methods for electroencephalography (EEG) analysis, highlighting variability in validation practices and reporting. They propose a Deep Learning-EEG (DL-EEG) checklist to help with reporting key algorithmic details, which shares some overlap with our proposed checklist.

In contrast, our work specifically addresses validation procedures and their direct influence on scientific conclusions. Proper validation is critical for ensuring reproducible results, preventing inflated performance estimates, and avoiding misinterpretation of findings. Note that a recent survey that found that many clinical neural engineering papers using EEG and deep learning for medical diagnosis were incorrectly splitting their training and testing data.¹⁵ While MOABB and similar initiatives provide valuable benchmarks for machine learning models, they do not offer generalizable frameworks for addressing the diverse challenges inherent in validating performance on neural engineering datasets. Our checklist fills this gap by offering a systematic approach to link validation strategies with scientific objectives, ensuring that research in neural engineering is reproducible, rigorous, and meaningful.

The rest of this paper is structured as follows. Prior to introducing our checklist, we begin with a review and tutorial of standard validation practices in Section 2. In this section, we will consider real-world examples in neural engineering and illustrate the effect of applying different validation techniques to real-world data. We will show how common validation errors lead to overstated performance and incorrect scientific conclusions. Then, Section 3 discusses other considerations for machine learning beyond predictive performance. This highlights that machine learning methods do not exist in a vacuum. They are part of a broader ecosystem of use, and their utility depends on how they are designed and used by both practitioners and (potentially non-expert) target users. Section 4 presents the NERVE-ML checklist in detail, highlighting its tailored approach to addressing the unique challenges of neural engineering, such as reproducibility, validation, and translational impact. Section 5 concludes our discussion.

2. Validation and Model Selection:

The validation process attempts to answer the question: how well will a machine learning system trained on a subset of data generalize to the entire population? The canonical approach in a data-rich scenario is to randomly split all data samples into training, validation, and test sets prior to analysis¹⁶, which is shown in Figure 1(a) where the data (samples corresponding to subjects or measurements) are randomly assigned to these sets. This approach is commonly used in large-scale datasets in deep learning, such as in the ImageNet challenge that helped advance deep learning techniques¹⁷; however, it is not always feasible in neural engineering, as we will highlight below. To briefly describe our nomenclature, the training data will be used to train machine learning models and learn the parameters in the machine learning system or model, the validation data will be used repeatedly to estimate performance of many models and perform model selection (meaning choosing which hyperparameters and/or model to use in practice), and the test data will be used only once to estimate population level performance. We note that the nomenclature is inconsistent in the machine learning literature, such as where “test” and “validation” have opposite meanings in different papers, or terms such as “holdout,” “blind,” “internal validation,” and “external validation” are used in place of either of these terms. This lack of standard nomenclature and overlapping jargon creates some challenges and hinders reproducibility, so we will note when multiple terms are commonly used for the same idea.

Prior to deploying a system relying on a machine learning model, we need to assess whether its performance is fit for the intended purpose. The *validation* procedure mimics the deployment process of the machine learning model on a new real-world experiment by applying it to data previously unseen by the learning algorithm (i.e., different than the training data, and independent from it). In other words, we are trying to see how well the model generalizes to new, previously unseen data. Theoretically, trying a machine learning model on such new data can provide a good estimation of real-world performance.

Therefore, the validation procedure also plays a critical role in selecting the best model (“model selection”) and hyperparameter tuning. It is generally not helpful to use classification performance on the training set to select models and hyperparameters. In fact, neural network methods often make zero training errors in classification problems¹⁸. This does not mean that they will perform equally well in the entire population. As such, we want to pick the best machine learning model and settings (or hyperparameters) by trying many possibilities and choosing the approach that will work the best on the true population. As the true population is inaccessible, we approximately find the optimal modeling approach by choosing the one with the highest performance on the validation data, which again is not used to train or fit the model. This procedure is shown in Figure 1(b). Because we try many different approaches on the validation data and there is randomness due to the sample size, our validation performance estimates become biased¹⁹. In other words, if we try enough methods, one of them will appear better than it actually is by random chance. Therefore, we need to use the previously unused test data to get an unbiased estimate of population-level performance, or what is often called generalization error. That approach ensures that no test data is ever used to optimize

the model (i.e. by tuning of its parameters or hyperparameters). This testing approach is shown in Figure 1(c).

This data splitting scenario is simple and works well when data samples are truly independent and plentiful. However, it is often inappropriate for neural engineering, as it is in many fields. We focus here on three main issues. First, how we split the data into training, validation, and test can change what scientific question we ask of the data depending on its structure. This is especially important as data samples in neural engineering are seldom completely independent, such as multiple samples or sessions from a single individual. Second, it is rare to be in a truly data-rich scenario in neural engineering, and alternative methods, described below, can be beneficial for estimating population-level performance by reducing uncertainty and making better use of all data. Third, there are potential mismatches between the research data and the true population, which is not a unique problem in machine learning.

2.1 Validation with Limited Data

Neural engineering research is heavily limited by number of subjects and practical limits on data collection. As such, it is important to consider how we can use more of the data for validation and model selection. The most standard approach for this is to use cross-validation¹⁶. Rather than using a predefined training and validation set, the idea of cross-validation is that we create a pool of combined data that is repeatedly used to create training and validation data. The most common approach is to use K -Fold Cross-Validation, where the data is split into K non-overlapping validation sets. For example, if $K = 3$, then three separate validation sets are formed. In each iteration, one of these sets is held out for validation while the remaining two are combined for training. After the model is trained on these two sets, its performance is ideally evaluated on the held-out data. This process is repeated K times, ensuring that each of the K subsets has served as a validation set. This process is visualized in Figure 2. The advantage of K -Fold Cross-Validation in neural engineering, with its limited data, is that it allows for more extensive use of the available data for both training and validation, thereby increasing the reliability of the validation process. Furthermore, it provides insights into the variability of model performance across different data subsets, highlighting potential biases or inconsistencies in the dataset.

There are multiple methods to get a final estimate of model performance. If cross-validation is run with test data held out, it means that a portion of the data is set aside right from the beginning and is not involved in the cross-validation process. This reserved test set would then ensure an unbiased assessment of model performance on unseen data. Such a strategy boosts confidence in the model's generalization capabilities since the final evaluation is carried out on data that had no influence on the model's training or hyperparameter tuning. However, this can be challenging in practice, as a 15% validation sample on 20 subjects is only 3 subjects. Given the typically large inter-subject variability in neural engineering, performance estimates will be highly uncertain. Such small numbers prohibit confidence in any performance numbers.

On the other hand, if cross-validation is run on all data, every data point is used in both the training and validation sets across different folds. While this approach maximizes the utilization of available data and may provide a more comprehensive view of model stability and

performance variability, it lacks a final evaluation on an unseen dataset. Hence, the predictive performance tends to be overreported, although methods such as nested cross-validation can mitigate this bias^{20,21}. Overinflated performances with cross-validation are often observed in neural engineering²². For instance, neural signals are known to be highly variable and non-stationary across time. However, with cross-validation, it is not uncommon that the training data includes some samples collected before (in time) those of the validation data, and some collected after it. As such, the training data can capture this variability, contrary to a real online neural engineering experiments in which the test set is always later in time than the training set. Moreover, performance estimates with cross-validation in neural engineering with small data can lead to unstable performance estimates with large error bars²³. Thus, especially in applications where deployment decisions are crucial, having an independent test set can be a safeguard against potential overfitting or other biases that might not be captured during the cross-validation process.

2.2 Statistical Questions and Data Splitting

In this section, we explore how different data splitting procedures relate to different scientific questions. Ideally, our research question will determine how we split the data, as the choice of splitting strategy directly influences the validity and interpretability of the results. The goal of this section is to clarify how variations in data splitting strategies impact the scientific conclusions we draw from machine learning experiments. Different splitting strategies, such as dividing data by subjects, sessions, or randomly across the dataset, are all valid but serve distinct purposes and should be aligned with the specific scientific objectives. By carefully selecting the splitting method, we can ensure that the evaluation metrics meaningfully address the research question at hand.

This discussion builds on prior work emphasizing the importance of validation practices in neural data analysis. For instance, Varoquaux et al.²³ highlight the biases that can arise from inappropriate cross-validation strategies, while Roy et al.² point out how mismatched validation approaches can lead to inflated performance metrics in EEG-based machine learning studies. Our focus extends these insights by explicitly linking data splitting choices to the validity and interpretability of scientific conclusions in neural engineering.

As a concrete example, consider a tool where we classify brain data on a second-by-second basis, as is a common task in seizure detection, brain-computer interfaces and emotion recognition. We assume that we have data from M subjects all from the same medical center, and each subject m has data split into N_m instances (e.g., one instance for each second) with features x_{mi} and targets y_{mi} ^b. These data may have been collected over multiple sessions. A visualization of this data structure for a single subject is shown in Figure 3, which shows a visualization of individual instances and sessions. We note that Figure 3 uses multi-channel

^b The terms used for the features and targets are not standardized. It is common to instead call the features “predictors,” “covariates,” or “input variables,” among others. Likewise, it is common to call the targets “outcome,” “label,” “class,” or “dependent variable,” among others.

electrophysiological data and binary targets as an example, but the concept generalizes to many different data types and targets.

To understand how different splitting approaches can relate to different corresponding scientific questions, consider the concrete example of seizure detection. We may want to ask:

- a. How well does the machine learning model predict targets on new instances from the same subject?
- b. How well does the machine learning model predict targets on a new session (e.g., another day) from the same subject?
- c. How well does the machine learning model predict targets on new subjects at the same medical center?
- d. How well does the machine learning model predict targets on new subjects at a new medical center?

The appropriate data splitting strategy is determined by the scientific research question. As highlighted above, the validation strategy should mimic a new real-world experiment aligned with that question. To answer question (a) with an experiment, we would collect more data from the same subject during the same session, and then evaluate the performance on these newly acquired instances. This can be mimicked in a validation procedure by randomly choosing instances, and then evaluating on previously unseen instances. This is visualized in Figure 4(a). This is a standard validation procedure commonly used in machine learning practice, and implicitly assumes that all data instances are independent. In this case, the model to be trained is referred as the subject-dependent model²⁴.

To answer question (b) with an experiment, we would bring the same subject back for another session, such as after a break or on another day. This helps to understand whether the predictions are consistent across sessions. This procedure can again be mimicked in a validation procedure by ensuring that all data from a given session stay in the same split of data. When performed with cross-validation, this is commonly referred to as *Group Cross-Validation*, where the randomization is over groups of data rather than over individual instances. We visualize how this process would be applied in cross-validation in Figure 4(b) compared to the instance-level split in Figure 4(a), which use the visualization of sessions from Figure 3. In the instance-level validation, each subject and session's data are in the training or testing set on every fold, and there is no structure to which data is held-out at any given point. Conversely, in the case where we split by session, all data from a complete session is either in the training or validation, where this creates structure on which data instances are in the validation or training set on each fold. Unsurprisingly, this is a more challenging task for a machine learning algorithm, which we show empirically below, as it requires an algorithm to achieve a greater level of generalization. In this case, the model to be trained is referred as the cross-session model²⁴.

To answer question (c) with an experiment, we would need to bring in new subjects that we had not included before in the experiment and try the algorithm on them in a prospective manner. In a validation procedure, we can mimic this experiment by using complete subjects for validation and test data rather than individual instances or sessions. This would estimate performance of our machine learning model on unseen subjects' data, which more closely mimics the motivating experiment and test in practice. This is again a type of *Group Cross-*

Validation, where the randomization is done by subjects. We visualize this procedure in Figure 4(c), which provides more structure than grouping over sessions. Subject-level validation is more challenging still as it requires further generalization compared with session-level validation within the same subject. In this case, the model to be trained is referred as the subject-independent or cross-subject model, e.g., in BCI for EEG-based emotion recognition or vigilance estimation²². However, this type of subject-based validation approach is appropriate for many questions in neuroscience⁷. For smaller datasets, these careful validation strategies are even more critical, as the limited data increases the risk of overfitting and inflating performance metrics.

Finally, we note that it is impossible to ask question (d) from this data without making significant assumptions, as our data all came from a single center. Evaluating models on another center is often referred to a form of external validity, and is a more challenging level of generalization. Such efforts require explicitly comparing and combining data from multiple medical centers. We include it here to highlight that there are always limitations of the data, and that it is important to address the limits of generalization of any method. We note that there are many scales of generalization that are useful in practice, and that a common goal of translational research is to gradually increase the scale of generalization. For example, in addition to the cases mentioned above, this could include generalizing across genotypes, measurement devices, or species²⁵, which is beyond the discussion here.

A natural follow-up question is how much these differences in validation matter in practice. Like most of statistics, the answer is that *it depends* on the data properties and question; for example, in the case studies below, we show that because electrophysiological signal can as electrophysiological signals may contain information about person-specific traits or conditions^{26–28}, targets that are constant for a subject across instances (e.g., features allowing diagnosis of mental disorders) require careful adjustment to account for the data structure. Roy et al.² highlighted similar discrepancies in EEG-based deep learning studies, noting that validation strategies can report highly discrepant values for intra-subject and inter-subject validation procedures. Similarly, the HAMLET framework²⁹ explored intra-subject and inter-subject validation settings, showing a huge discrepancy in performance between these two cases and demonstrating the challenges of generalizing across subjects. Thus, different validation practices can make a huge difference in practice in our performance estimates under common conditions, often rendering unrealistically optimistic performance estimates.

Focusing on validation practices is essential to explicitly demonstrate how methodological choices impact reported results. Below, we directly examine how validation strategies on several real-world datasets relate to research questions and interact with data structures. By showcasing the exact impact of different cross-validation procedures on performance estimates, we hope to provide researchers with practical insights to design more rigorous and relevant validation strategies.

2.2.1 Animal Model of Stress Response

First, we will look at a task related to brain state, which relates local field potentials collected in mice from implanted electrodes in 11 different brain regions to stress and genotype. This dataset is publicly available^{30,31}. This data was previously used to help design a

neurostimulation protocol to mitigate the impact of stress, as described in full elsewhere³². Briefly, the data consists of 14 wild-type mice and 12 Clock-Δ19 mice. The Clock-Δ19 genotype is a mouse model of bipolar disorder³³. Each mouse has a 300 second recording in their home cage, a 300 second recording in an open field test, and 600 second during a tail suspension test. The tail suspension test is used to assess response to a challenging experience³⁴. We would like to discover the brain signature relating to stressful conditions and to genotype. We therefore set up the machine learning problem where x_{mi} represents the derived features for the i^{th} second of the m^{th} mouse. Here, there are $M = 26$ subjects and $N = 58135$ total instances and data was collected in a single session for each subject. These features represent the frequency-based power and frequency-based coherence between brain regions, resulting in 3696 different features³¹. We then define the corresponding target y_{mi} as either the stress condition (open field and tail suspension are considered stressed and home cage as non-stressed) or genotype (wild-type versus Clock-Δ19).

For our purposes here, we choose a standard multi-layer perceptron with a single hidden layer of 10 hidden units with a Rectified Linear Unit ("ReLU") nonlinearity, as neural networks are very standardly used models in the literature. We are not performing model selection (e.g., only using one configuration) but rather focusing on the differences in performance estimates due to the validation procedure. It is unlikely that this is an optimal modeling approach; however, our focus here is on the impact of the validation on a reasonable machine learning procedure rather than finding the optimal predictive model.

To understand the impact of the validation procedure and the importance of carefully choosing the validation procedure, we explore the impact on performance estimates from several different validation procedures where we predict both stress and genotype. First, we show the result from not doing validation, i.e., evaluating the classifier on the training data. Not doing validation results in inappropriate results, and we only include it here as an example of a bad practice that has been responsible for previously retracted papers³⁵. Second, we perform a 5-fold cross-validation over instances, which is the strategy implied by question (a) above, i.e., classifying unseen data from the same subject. Third, we use the 5-fold group cross-validation approach over mouse identity, which is the strategy implied by question (c) above, i.e., classifying unseen data from unseen subjects.

We report performance on these binary tasks by using Area Under the Receiver Operating Characteristic Curve (AUROC, usually simplified to AUC). AUC is a standard machine learning metric where 1.000 is a perfect classifier, and 0.500 represents classification no better than chance (e.g., no information). The results are shown in Table 1. Succinctly, not performing cross-validation results reports near-perfect performance, but it is dramatically incorrect. It may be surprising that the result from not doing validation is essentially perfect, but neural networks routinely overfit to this extent on training data despite reasonable generalization³⁶.

Next, we note that the performance from the instance-based validation still performs very well, but there is a significant discrepancy between the by-instance K-fold and by-subject K-fold for both binary tasks. This discrepancy is practically important as the by-subject validation strategy matches the relevant scientific question here, which is how well we could identify a stress condition in a new subject. In the stress prediction case, this discrepancy would imply roughly

twice as many errors as predicted by the instance-based validation if the method was applied to a new subject, which is more realistic. Furthermore, the gap between the performance in predicting genotype reported by these two different validation approaches is gigantic. This neural network looks almost perfect in the instance-level validation, whereas the by-subject validation is barely greater than 0.500 with significant uncertainty. We use the standard deviation as a simple measure to convey uncertainty as constructing confidence intervals on predictions from cross-validation data is complex due to the dependence between the folds³⁷.

This result may at first seem surprising, but it highlights the limitations of the collected data. While the number of instances ($N = 58135$) seems like it is “big data,” we are statistically limited by the total number of subjects ($M = 26$). This is compounded by the uniqueness of brain signatures. In fact, brains are considered as a potential biometric because of their unique signals and responses^{26–28}. To demonstrate this effect, we next evaluated whether this same neural network architecture could be adapted to identify the subject from instances of electrophysiological data. This requires adapting the neural network to predict one of many classes rather than a binary through a small coding change. We found that without a validation, it seems again essentially perfect. When using instance-level validation, we found that it could identify the source subject out of the 26 possibilities most of the time (95%), meaning that subject identity is embedded in the neural data. We do not provide group K-fold results here, as the performance of this model architecture to predict new subjects is very poor.

This leads to an important conclusion: when predicting subject-level data, the instance-level validation strategy can lead to the classifier just memorizing the subject’s identity, which can confound interpreting subject-level performance. To emphasize this point, we next randomly assigned each subject a binary (artificial and random) label and evaluated how well the neural network could predict these arbitrary labels. This procedure is visualized in Figure 5. Here we show that each subject is assigned a random binary target, and each instance in that subject gets the exact same target. As expected, without using a validation set, the AUC implies perfection. Performing validation over instances, though, is misleading, as it reports a stunningly high AUC of 0.979, which is essentially perfect. The neural network accomplishes this task by simply memorizing which subject is which, as the labels have a completely random relationship with the neural data. When validating by subject, the performance estimate clearly overlaps with a 0.500 AUC, so there is no significant evidence that the neural network can truly predict this relationship. It may be surprising that the by-instance validation did even better with a random label than it did predicting subject identity; however, it is worth noting that to predict a random label, you only need to identify a group of subjects rather than an individual, which is an easier task. Second, the fact that the by subject validation was above 0.500 for random targets may be surprising, but this is simply due to random chance with “small” data. With enough independent repeats of this experiment, we would expect that the hold out AUC over subjects would be exactly 0.500.

2.2.2 Emotion Recognition

In this section, we explore the task of emotion classification using EEG signals. This analysis involves EEG data recorded from 16 healthy participants who watch video clips rich in emotional content across three sessions. The data used for this task is the SEED-V dataset, as detailed in

previous works^{24,38,39}. During the experiments, participants watch 45 film clips, confirmed through prior studies to consistently evoke specific emotions—happiness, sadness, disgust, neutrality, and fear. The dataset is partitioned into three sessions, each containing 15 clips (trials) to ensure an even distribution of three clips per emotion, thus maintaining a balanced representation of emotional stimuli. We follow data preprocessing and feature extraction as detailed in previous work³⁸. Succinctly, the raw EEG data are initially downsampled to a 200 Hz sampling rate. This is followed by the application of a bandpass filter ranging from 1 Hz to 75 Hz to reduce noise and artifacts from the EEG data.

For feature extraction, we use the differential entropy (DE) features from the EEG signals across five frequency bands—delta (1-4 Hz), theta (4-8 Hz), alpha (8-14 Hz), beta (14-31 Hz), and gamma (31-50 Hz)—across all 62 channels, resulting in extracted features of 310 dimensions (62 channels $\times$ 5 frequency bands). The machine learning framework is therefore set up with each feature matrix for each trial, session, and subject, and each of which has its own target emotional class. Here, there are $M = 16$ subjects, 3 sessions and 15 trials per subject, and an instance is considered every 4 seconds of non-overlapping brain data. This results in $N = 29168$ total instances. For modeling, we choose a standard multi-layer perceptron with a single hidden layer of 256 hidden units with a Rectified Linear Unit (“ReLU”) nonlinearity. As mentioned in Section 2.2.1, it is unlikely that this is an optimal modeling approach; again, our focus here is on the impact of the validation procedure rather than finding the optimal predictive model.

To understand the impact of the validation procedure and the importance of carefully choosing the appropriate validation procedure, we explore how different validation procedures affect performance estimates in predicting emotion class. First, we show the result from not doing validation. Again, not doing validation results in inappropriate results. Second, we use a K-fold cross-validation over instances, which is the strategy implied by question (a) above. This method, although common, may not be ideal for our dataset where multiple instances from the same trial correspond to the same emotion class. Such overlap can lead to overfitting as “new” instances may not be truly independent, having potentially shared information with other instances within the same trial. In other words, some instances from one trial may be assigned to the training set, while other (potentially overlapping) instances from the same trial may be assigned to the test, preventing the training and test sets from being independent. We then employ a K-fold group cross-validation strategies that consider different aspects of our dataset's structure: subject identity, trial identity, session identity, and combinations thereof.

For session identity, we use 3-fold cross-validation reflecting the three available sessions, and 5-fold cross-validation for all other identities, representing different and increasing levels of generalization^{24,38,39}. The results reporting prediction accuracies are shown in Table 2. Our results reiterate that not performing cross-validation or using standard K-fold cross-validation often yields misleadingly high performance. These methods fail to test the model against true new instances, highlighting the critical importance of selecting appropriate validation strategies for accurate performance evaluation. Next, we observe a significant performance drop when using K-fold group cross-validation. Specifically, using K-fold group cross-validation by subject

identity, we find that accurately identifying an emotion class in a new subject is markedly challenging, with an accuracy of only 0.318. This result contrasts with those obtained using K-fold cross-validation by instance, highlighting the importance of selecting the appropriate cross-validation strategy.

As highlighted in Section 2.2.1, despite having large number of instances ($N=29,168$), our statistical power is significantly constrained by the number of subjects ($M=16$). This is compounded by the uniqueness of EEG signals^{40,41}. To demonstrate this effect, we test whether the same neural network architecture could identify subject, trial, and session identities using the provided differential entropy (DE) features from EEG signals. We observed that without any form of validation or using standard K-fold cross-validation, the model appears to perform perfectly. Interestingly, when using group K-fold cross-validation by session or trial, the model is still able to accurately identify the source subject and session identity nearly perfectly and above chance, respectively, indicating that these identities are inherently encoded in the EEG data. This finding further supports the conclusion that instance-level and some group-level validation strategies may memorize subject and session identities rather than learning to generalize.

To further demonstrate the robustness of the different cross-validation procedures, we randomly assign one of the five targets to the data and assess the neural network's ability to predict random targets when they are structured either by subject, where all instances of a subject's data are randomly assigned the same target, or by session, where all instances of a session are assigned the same target. The first case is aligned with the prior conclusions above, where only cross-validation at the subject level captures the fact that it is a random relationship. Notably, even splitting by session gave an accuracy of 93.7%, whereas splitting by subject has an accuracy of 20.9% with high uncertainty (compared to a theoretical expectation of 20%). When randomly assigning targets by session, we see trends suggesting that we are not necessarily capturing the relevant information when session is related to target structure. Here, the network can estimate the correct target 38.4% of the time in cross-subject validation, whereas it could estimate the correct session 39.7% of the time. In other words, the network is capturing the structure related to session, not target. This further emphasizes that we need to consider structure of the targets and the research goals when choosing the validation procedure.

2.2.3 Sleep Staging

In this section, we explore the task of sleep stage classification using EEG signals to explore the properties that occur when the targets have natural, rather than enforced, structure through time. We classify 30-second segments of polysomnography (PSG), referred to as "epochs," into different sleep stages using the Sleep Cassette (SC) subset of the 2018 Sleep-EDF dataset, also known as SC-EDF-20⁴². This dataset includes scalp EEG recordings from two channels (Fpz-Cz and Pz-Cz) across 20 healthy participants. Following established protocols^{43–49}, we consider only the PSG data from 30 minutes before to 30 minutes after the recorded sleep period, as the dataset includes a lot of wake data outside this window. The epochs are scored using the Rechtschaffen and Kales (R&K) rules, with adaptations to align with the American Academy of Sleep Medicine (AASM) standards by merging stages N3 and N4 into a single N3 stage. We exclude epochs labeled as MOVEMENT or UNKNOWN. We adhere to the preprocessing and

feature extraction methods detailed in prior research⁴³, resulting in features of 1081 dimensions derived from both time and frequency domains, across multiple window sizes. Each feature vector, x_{mi} , represents the derived features for the i^{th} recording of the m^{th} subject. Here, there are $M = 20$ subjects and $N = 42230$ total instances. We then define the corresponding target y_{mi} as the sleep stage, categorized as wakefulness (W), stage N1, stage N2, stage N3, and rapid eye movement (REM). For the model, we use a standard multi-layer perceptron with two hidden layers containing 256 and 10 units respectively, each using a Rectified Linear Unit (ReLU) nonlinearity. Again, it is unlikely that this is an optimal modeling approach; our focus is on evaluating the validation procedure's impact rather than on identifying the most effective predictive model.

In this section we evaluate the impact of different validation procedures on the accuracy of sleep stage prediction. Firstly, we reiterate that neglecting to use any validation can lead to unreliable results. Secondly, we use a K-fold cross-validation over each instance, as outlined by question (a). Thirdly, we adopt a K-fold group cross-validation by subject identity in accordance with question (c). Beyond these methods, we also evaluate how much the temporal patterns in sleep labels impact predictions by doing a time K-fold cross-validation, illustrated in Figure 6(a). Validating by time is conceptually similar to by-session with the distinction that it is not necessarily following a session structure, but perhaps a single long session. It is commonly used in practice in evaluating sleep⁵⁰. In this approach, instances for each subject are partitioned into K contiguous segments. The models are then trained on four segments and tested on the fifth to measure performance, simulating the chronological sequence found in real-life data streams. This method not only addresses the strategy of question (a) but also investigates the significance of instance order in validation procedure selection. The results reporting prediction accuracies are shown in Table 3. Consistent with our earlier findings in Sections 2.2.1 and 2.2.2, not performing cross-validation results in perfect performance, but it is misleading due to the inherent properties of neural networks.

Next, we observe that instance-based validation performs well; however, a notable performance gap exists between instance-based K-Fold (Acc: 0.891) and the subject-based Group K-Fold (Acc: 0.827) as well as Time K-Fold (Acc: 0.831) approaches. These gaps are significant since the objective is to accurately identify sleep stages in new subjects, as well as to maintain accuracy within the same subject across a chronological data sequence that mimics real-life situations. To rigorously assess the difference between instance-level and subject-level validation, we applied the corrected resampled t-test¹⁰⁹, together with confidence intervals. This method accounts for the dependence between cross-validation folds, which should be considered when dealing with repeated measures in neural data. Here we perform a single round of 5-fold cross-validation ($k=5$) and estimate the correlation between folds as $1/k$. This approximation serves as a straightforward way to consider the inherent dependencies in cross-validation arising from shared training data.

Our analysis shows that instance-level K-fold cross-validation, while achieving a higher average accuracy (0.891 ± 0.0031), is overly optimistic because it allows the model to learn subject-specific characteristics from the training data. In contrast, subject-level K-fold cross-validation

requires the model to generalize to unseen subjects, resulting in a lower mean accuracy (0.827 ± 0.029) that more accurately reflects the challenge of generalizing across different individuals. The corrected resampled t-test shows a statistically significant difference between these two approaches, with a mean performance difference of 0.064 [95% CI: 0.013, 0.115], with the instance-level approach yielding significantly higher performance estimates ($p = 0.025$). The confidence interval is based on the mean difference between the two approaches and its corrected variance for fold dependencies.

Again, our model is hindered by having only 20 subjects ($M=20$), especially considering the heterogeneity of EEG signals across individuals^{40,41}. To further investigate this within the SC-EDF-20 dataset, we next evaluate whether this same neural network architecture could be adapted to identify the subject that provided the instance of sleep EEG data. Again, we find that without validation, it is almost perfect. Instance-level validation also demonstrates near-perfect performance in identifying the source subject, verifying that subject-specific characteristics are discernible in the EEG data. However, a notable decrease in accuracy is observed with Time K-fold cross-validation (Acc: 0.7716), suggesting that while the EEG data contains subject-specific information, the chronological structure of the data also influences model performance. As Section 2.2.1, we do not perform group K-fold across subject, as it is impossible in such a setting to predict new subjects.

This leads to an important conclusion: when predicting subject-level data, the instance-level validation strategy can lead to the classifier just memorizing the subject's identity. Following the approach in Sections 2.2.1 and 2.2.2, we assign a single random sleep stage labels to each subject and assess the neural network's predictive accuracy, meaning that each subject is assigned a single random, artificial label. Unsurprisingly, the model implies perfection on a validation set. Based on our enforced random label-data relationship, we would expect our model to generate predictions close to random. However, the high accuracy (0.969) is deceptive for instance-level validation, as it results from the model remembering subject identity rather than learning from the label-data relationship. Subject-based validation yields low accuracy (14.1%), indicating there is no significant evidence that this neural network could truly predict this relationship. Interestingly, instance-level and Time K-Fold cross-validation performed better with random labels than in identifying subject identity. As discussed earlier, this may be because these methods only need to identify a group of subjects rather than an individual, which is inherently less complex.

Next, we assess whether a model is learning meaningful patterns from the data that generalize across individuals, or if it is simply memorizing the data-label pairing for each subject by memorizing points in time. To evaluate this, we apply a label shuffling procedure among the subjects within our dataset, as shown in Figure 6(b). For example, we randomly reassign subject k 's labels to subject j 's data, thereby preserving the temporal structure of the data but deliberately disrupting the inherent link between the data and its corresponding target. Due to the variation in the number of instances per individual, we restrict our analysis to subjects with over 2000 instances. Despite the shuffling, instance-level validation still has 58.8% accuracy over 5 classes, meaning that our model captures noisy temporal patterns that do not depend on the

actual label relationships. These results emphasize how critical it is to match the scientific question to the splitting procedure.

2.2.5 Considerations for Dataset Size

The validation strategies discussed in this section are influenced by the size of the dataset, as different data sizes require adjustments to ensure robust and reliable results. This is especially apparent in the case studies presented earlier, where we are statistically limited by the number of subjects, which varied from 16 to 26. The classification performance gap we observed between splitting strategies in these case studies is designed to convey how impactful careful selection of the scientific question and subsequent cross-validation procedure are on statistical properties. Techniques such as nested cross-validation (see Section 2.3) can be particularly useful in these cases.

For larger datasets, the performance gap between different validation strategies is expected to decrease but remains a critical consideration. For instance, in a recent study identifying an EEG biomarker for subjects diagnosed with autism compared to neurotypical controls, we analyzed a dataset with 293 total subjects⁵¹. Using subject-level cross-validation, the model achieved an AUC of 0.72. However, when evaluated instead using instance-level validation paired with an expressive deep learning method, the AUC exceeded 0.9. This discrepancy highlights the importance of specifying the correct validation procedure, as improper validation can lead to overly optimistic performance estimates (here the AUC of 0.9), that would not reflect real performances on unseen subjects.

For large datasets, careful consideration of computational demands is crucial, as different validation techniques can significantly impact overall costs. Efficient sampling methods may be necessary to reduce computational burdens while preserving the integrity of the evaluation. In deep learning, for example, it is common practice to use a single validation split rather than cross-validation to balance these trade-offs. By explicitly addressing such considerations, we can ensure that validation strategies remain both scalable and applicable across diverse neural engineering research contexts.

2.2.6 Considerations for Temporal Structure:

Addressing the inherent temporal structure within neural data needs careful consideration at both the data preprocessing and model selection stages. When partitioning data for validation purposes, it is important to use methods that consider temporal dependencies. One effective method involves partitioning the data into contiguous temporal blocks, ensuring that data points within a defined block stay in the same cross-validation fold. This approach, analogous to the validation strategy previously discussed in the context of sleep stage classification, is essential to prevent data leakage between training and testing sets, which can artificially inflate performance metrics. Furthermore, data organization can be structured around temporal components, such as grouping data within sessions or subjects. This systematic grouping ensures that samples exhibiting similar temporal dependencies are collectively managed, maintaining the integrity of the temporal relationships inherent in the data.

In addition to data splitting considerations, the selection of appropriate modeling techniques is important for effectively capturing temporal dynamics, including many different statistical and

deep learning models. These models possess the capacity to learn from the temporal context embedded within neural signals, thereby enabling the capture of both short-range and long-range temporal dependencies. Such approaches can make a large impact on predictive performance and scientific conclusions.

2.3 Challenges in Model Evaluation

The previous section dealt primarily with the challenges in interpretation due to a variety of ways of data splitting within cross-validation. As can be seen, estimates of performance can highly vary due to the chosen procedure, and it is important to link the scientific question to the procedure for performance to hold up in practice, and for the machine learning classifier to learn and use task-relevant patterns, and not confounding patterns, e.g., patterns related to subject identity. Note that the latter issue was recently noted in a survey that found that many clinical neural engineering papers using EEG and deep learning for medical diagnosis were incorrectly splitting their training and testing data.¹⁵ This makes it rather likely that these classifiers were (incorrectly) using the subjects' identities for classification rather than disease-related EEG features, thus vastly overestimating the diagnosis accuracy.

While we report predictive performance by averaging over the different validation folds in the cross-validation procedure, there are underlying biases to address in this procedure. A downside of using such a cross-validation procedure to report performance is that it will overreport performance when used to select from many different models (e.g., if you compare many models, one of them may look good by random chance)²⁰. In fact, despite the common knowledge that machine learning models improve with more data⁵², the reported performance in the literature can decrease with increasing N as the bias from the cross-validation procedure decreases⁵³. In other words, even though the model is getting better, it could appear worse because the upward performance bias from model selection during the cross-validation decreases.

Historically, a variety of methods have been used to try to improve validation efforts. For example, nested cross-validation²⁰ has been commonly used to mitigate performance reporting bias and estimating confidence intervals³⁷. In nested cross-validation, the model selection procedure has two levels of cross-validation. The first level is the same as in typical cross-validation, and the second level is performed within each cross-validation split. Model selection is performed on the second level within each first level split, and then the first level split is used to estimate performance. This technique has been shown to report more accurate performance. Additionally, it is relatively common to run permutation tests⁵⁴, which can be used to estimate what type of predictive performance we might expect with random information. We demonstrate this idea above where we swap the labels between subjects in the sleep data and evaluate their performance but used as a statistical testing procedure. This allows us to evaluate whether the underlying structure impacts our predictive ability. Additionally, it is possible to perform statistical analysis based on the distribution that comes out of repeated permutation testing. We emphasize that when doing permutation testing it is critical to choose which structure is kept in the permutation test (e.g., do we permute over instances, sessions, or subjects), as highlighted in the case studies above.

For a test set with little data, which is often the case in neural engineering, the chance level performance metric should be reported and compared to the actual performance obtained. This is analogous to some of our previous experiments, where we use random targets and permutation tests to evaluate chance performance. For classification accuracy, there are analytical estimates of the chance level⁵⁵. For other performance metrics, or for regression problems, chance level performance should also be reported and could be empirically estimated using permutation tests, as described above.

We also emphasize that there should be careful consideration of the metric chosen to evaluate and compare performance. Above, we choose to largely use accuracy and AUC to illustrate the examples as it is a common metric that is easy to understand and show the behaviors of the different cross-validation techniques. However, it is also common to use metrics based on precision-recall, along with a host of metrics appropriate for comparing modeling performance or the utility of additional information⁵⁶. For unbalanced test data, i.e., with a different number of instances in each class, a performance metric that can handle imbalance (e.g., AUC, balanced accuracy, precision-recall curves, F1 score, etc.) should be chosen²². Classification accuracy does *not* naturally handle class imbalance, and is thus not suitable to estimate performance on unbalanced data. This metric should be chosen in careful consideration with what the scientific goal or question is. These metrics can be adapted for proper statistical testing, which is relatively straightforward on a true held-out set and requires more complicated procedures under cross-validation estimates^{20,37}.

2.4 Increasing Scales of Generalization

Different ways of collecting data and splitting cohorts seek generalization at different scales of the data. For example, we can think about generalizing to future data from the same subject, a new subject, a new population, a new medical center, or a different type of device, etc^{25,57}. Assessing these different scales of generalization requires different data collection efforts and processing, and as highlighted above requires different validation procedures. When venturing into new research areas, it is acceptable and often advisable to begin with investigations that have narrower or smaller levels of generalization. This is because pioneering research topics might not have sufficient data or prior knowledge to make broad generalizations immediately. Instead, focusing on specific cases, subpopulations, or conditions can provide valuable insights and set the foundation for future, more expansive studies. For example, neuromotor interfaces are gradually advancing from individualized computational methods to generic interfaces that work across subjects⁵⁸. However, as the field continues to move forward, it is crucial that researchers explicitly communicate the scale of generalization that they are evaluating and operating at in their studies. They should delineate the bounds of their findings and clarify how these findings and the validation procedures apply within those bounds. This ensures that the research results are not misinterpreted or wrongly extrapolated to broader contexts.

Furthermore, linking the chosen scale of generalization to the overarching scientific question helps justify the chosen scope and reinforces the study's relevance. By being transparent and intentional about the scope of generalization and how it relates to the chosen validation

approaches, researchers can build credibility, foster a clear understanding of their work, and pave the way for successive research that expands on these initial findings.

2.5 Generalization in Deep Learning

In most machine learning systems besides deep learning, the balance between overfitting and validation is carefully monitored, with the expectation both that the validation performance is reasonable but also close to the estimated performance on the training data¹⁶. In other words, we have balanced overfitting well, so the model performs similarly on the training and validation data. This principle is also echoed in many statistical models. However, deep learning deviates from this norm. In most real-world scenarios, a well-performing deep learning model has training error that is essentially zero¹⁸. This fact underscores the importance of focusing on the validation error independently, irrespective of an apparent overfit on training data, and motivates the checks outlined in this manuscript.

One might wonder how deep learning models, despite having numerous parameters, can fit limited data so well. Remarkably, neural networks can still generalize well even if they achieve flawless predictions on the training set. There are ongoing theoretical studies to understand the learning mechanics of artificial neural networks⁵⁹ and how this fact arises. This phenomenon is further elucidated by the concept of “double descent”⁶⁰, where a deep learning model’s generalization can improve as the number of parameters increases. This scenario is in deep contrast to typical intuition, where increasing the parameterization gives rise to more complicated and more overfit models. These nuances reinforce how robust and dependable validation is necessary in neural engineering applications.

2.6 Current Research in Advanced Machine Learning Techniques

Machine learning and artificial intelligence is rapidly advancing, as can be readily seen by the clear advances in computer vision and large language models over the last few years. Many of these advanced techniques are being adapted to tasks of interest in neuroscience and neural engineering. Recent techniques can be used to effectively improve machine learning performance on tasks of interest; while we encourage the use and further development of modern techniques, we emphasize that it is important to consider how these techniques may alter the assumptions of the data analysis and therefore change the scientific question.

As a concrete example, we will highlight unsupervised domain adaptation⁶¹. This has been applied by several groups on neural engineering tasks^{62,63}, with the general goal of improving how well we can predict labels on a new subject or patient that does not have ground truth labels. The key assumption of such techniques is that we have *unlabeled* data available for a new subject, meaning for example that we have access to the brain recording of a new subject during the training process. We further note that in the case where machine learning systems are previously pretrained and used in transfer learning, we cannot even guarantee that the new user has not been included in the training data in general, and care should be taken in making such assumptions. In this situation, this methodology can adapt and analyze the difference between training subjects and the validation subject and improve test performance. While this task can exist in practice (or models can be retrained), it changes the scientific question from “How well does this method work on a new subject?” to “How well does this method work on a

new subject where we retrain with access to the brain recording but not targets?" Further extensions of these ideas can be more directly relevant to the first question, such as zero-shot learning or domain generalization⁶⁴, but our key point is that we need to carefully consider how the training and assumptions of the methodology changes the interpretation of the results.

Additionally, other modern techniques to automate the model search (e.g., neural architecture search⁶⁵, Bayesian optimization⁶⁶) can help identify the most promising models and decrease human effort but may exacerbate overreporting results if only a validation set is used by increasing the number of comparisons. These challenges can be addressed by the statistical methods in Section 2.3. On this topic, it may be worth reminding that the selection of machine learning models, hyperparameters and features should be made using only training data (possibly using inner cross-validation on the training data), and not according to the final independent test set performance, nor using all the data together. Selecting them according to classification performance on data including the test set may result in vastly overestimating such performance, including reaching high accuracy even on completely random data without class information⁶⁷. For instance, the architecture of a Deep Neural Network should not be selected according to which one reaches the best accuracy on the test data, but rather the validation data. Likewise, when doing feature selection prior to model training, the input features should not be selected according to which ones are the most discriminant on all the data available, but rather only on the training data to properly estimate performance. Both these examples are commonly seen mistakes, included in some published papers⁶⁸.

Finally, we note that large-scale pretraining through self-supervised learning, which have been so prevalent in computer vision and natural language processing, are increasingly becoming feasible in many neural engineering tasks for electrophysiology^{69,70} and are already widely used for computer vision behavior tracking⁷¹⁻⁷³. These techniques have preliminarily demonstrated the potential to help improve general representation of physiological signals and generalization of machine learning models in neural engineering tasks^{69,70,74,75}.

3. Considerations Besides Validation and Generalization

When considering a machine learning model, model selection is typically based strictly on predictive performance. However, there are many other considerations when choosing and deploying a machine learning model. While detailed discussion is beyond the scope of this manuscript, we will briefly discuss a few common additional considerations and their relevance in neural engineering.

3.1 Interpretability and Explainability

Deep learning is by default considered a black box, meaning that it is indecipherable how the algorithm decides on the predicted outcome^{76,77}. In neural engineering, the use of black box methods like deep learning can be problematic when making high stakes decisions about brain-machine interfaces or neuroprosthetics, as clinicians and engineers may not understand how the model arrives at its conclusions⁷⁸. This lack of explainability can pose risks, especially if the model behaves unexpectedly during real-time neural signal processing or interventions (e.g., suppose data is input incorrectly from a system, but that error cannot be identified because it is

a black box). Furthermore, emerging regulation of AI-based systems impose requirements of transparency and explainability in certain applications, including the medical domain. Hence translational research efforts in neural engineering should carefully consider the use of transparent and explainable approaches.

In high-stakes decisions, it is common to focus on trying to open the black box and fully understand how the machine learning method is working. This is commonly done by considering interpretable methods or explainable artificial intelligence (XAI) ⁷⁹. Interpretability and explainability are sometimes referred to analogously but refer to two distinct concepts. Interpretability describes the degree to which a human can understand the procedure an algorithm uses to generate a result. A decision tree is a clear example of an interpretable model, where each individual step of the model can be understood and transparently communicated. Many other interpretable or “glass-box” models exist ⁸⁰ and have seen use in high-stakes applications such as high-profile examples in seizure prediction ⁸¹. It is worth noting, however, that these tools may still require significant effort and training to properly interpret ⁸². Furthermore, how the features are defined is a key consideration, as model results will be in terms of those features. The design of features in a way that carefully considers how preprocessing impacts scientific interpretation is an active area of research ⁸³. For interpretation, is it often ideal to use well-validated features, such as common clinical features ⁸¹.

Explainability, in contrast, usually focuses on post-hoc explanations of complex models. For example, we can visualize feature importance in a random forest or highlight sections of an image used in a convolutional neural network. Explainability methods have found significant uses in neural engineering applications. For example, we can explain how changes in neural signals relate to depression ⁸⁴, or develop brain networks of depression ⁸⁵ or social appetitive behavior ⁸⁶ that can be used to design neurostimulation protocols. However, such approaches can be inappropriate for high-stakes decisions, as the explanations are not always faithful to the actual mechanisms used to make decisions ⁷⁷.

Validation strategies can further influence interpretability and explainability. For instance, in subject-level validation, feature importance scores reflect patterns generalizable across subjects, making the results more robust for clinical interpretation. Conversely, in instance-level validation, feature importance may emphasize idiosyncratic subject-specific traits, leading to explanations that are less generalizable to new subjects or contexts. Both approaches can be appropriate depending on the research question at hand.

These validation strategies are also critical for uncovering biotypes⁸⁷ or patterns shared by a subset of subjects. For example, subject-level validation can identify features that are consistent across specific subgroups within the data, helping to define shared mechanisms or characteristics that underlie a particular condition. By contrast, instance-level validation may still provide insights into personalized patterns and potential responses to interventions, which can complement subgroup analysis.

XAI can also be a useful tool to assess whether a Deep Learning (DL) model is biased by confounding factors. For instance, in a motor imagery BCI experiment, Trocellier et al.⁸⁸ showed that including EEG data immediately following a visual instruction cue as training input features

led to a biased DL classifier. Indeed, such a classifier would use brain responses to that visual cue, rather than motor cortical activity, which also leads to inflated classification performance, and to an unusable classifier in practice as such an instruction cue is absent in real BCI use. This was discovered using XAI tools, which revealed that this DL model mostly used EEG signals immediately following the cue, in the brain occipital cortex. Removing EEG signals following the cue from the DL input, led to a network with a lower accuracy but using as features EEG signals during the motor imagery task, in the motor areas of the brain. Altogether, this also stresses the need to carefully control for possible confounding factors, e.g., task-related stimulus, when using black-box machine learning models in neural engineering, as well as to use XAI with DL to confirm the absence of such confounds.

3.2 Computational Constraints

While large scale machine learning is well-known to be data hungry, these models also demand significant computational resources⁸⁹. In many scenarios, trading in a minor performance gain in favor of an algorithm that is more computationally efficient improves usability. For instance, brain-computer interfaces (BCIs) used for motor rehabilitation or communication for paralyzed individuals require rapid and efficient processing to provide real-time feedback to the user. Similarly, neuroprosthetics, which assist patients in regaining sensory or motor functions, depend on swift and efficient algorithms to mimic natural neural responses. Another example can be found in wearable EEG systems used for seizure detection; these devices must operate efficiently to provide timely alerts without draining the battery. Likewise, closed-loop neurostimulation based on current brain patterns is dependent on minimal lag⁹⁰. As the neural engineering field continues to evolve and embrace more portable, wearable, and implantable devices, it is necessary to ensure computational efficiency of machine learning algorithms. This is especially important in the context of whether an algorithm is feasibly an online or real-time algorithm, a crucial consideration in many applications.

In addition to processing time, for classifying new data, the time and computational resources needed to train the machine learning models should also be considered. Notably, given the large variability of neural signals, both across and within users, many machine learning models in neural engineering are currently subject-specific and/or require frequent re-training to track such variabilities^{91,92}. For those models, the time needed to (re)train the model should be as short as possible, typically a couple minutes maximum, as during such (re)training, the user is waiting to actually use the system. Therefore, algorithms that requires hours or days to be trained would not be suitable in such real-life use situations.

The chosen validation procedure significantly impacts training time, particularly in cross-validation compared to a single validation split. For instance, in k -fold cross-validation—where k is the number of folds and N is the total number of data points—the computational complexity scales linearly with both k and N . However, k has a more pronounced effect, as the model must be trained k distinct times, and the number of folds is a parameter directly selectable by the researcher. Doubling the number of folds effectively doubles the total computation time. This scaling applies to both subject-level and instance-level splits, though the per-fold training cost may vary depending on the split type.

Additionally, the complexity of the chosen machine learning method greatly influences computational requirements. Neural networks, for example, have training costs determined by factors such as the number of parameters, the depth of the architecture, and the number of training epochs. More complex architectures, such as transformers, typically incur higher per-sample training costs than simpler models. Therefore, in neural engineering applications, selecting an algorithm that balances model complexity, training time, and computational efficiency is critical to achieving real-time responsiveness and overall usability.

3.3 Fairness

Machine learning models, by their very nature, are heavily influenced by the data on which they are trained. Language models, for instance, inherit biases from their human-created training data, leading to significant and sometimes problematic biases in real-world applications⁹³. While neural engineering might seem removed from such human-centric biases, data cohorts can still be skewed and non-representative. Academic research experiments are often conducted on cohorts of volunteer students that may not represent the characteristics of the target population in terms of their age, gender, ethnicity, among others. Data quality may not be uniformly high-quality across all groups of individuals. For instance, EEG devices have been observed to produce different results based on varying factors like hair type and scalp thickness⁹⁴, while near-infrared spectroscopy has been shown to often provide lower data quality on subjects with darker skin⁹⁵.

In medicine more broadly, it is becoming common to run a series of checks for potential machine learning model bias⁹⁶. These checks can likewise be applied in neural engineering applications when the total number of subjects is high enough to reliably estimate such differences. Recognizing these potential biases early allows researchers to address them proactively, either by refining the algorithms, recommending more balanced data collection, or implementing other bias-mitigation strategies. One approach to this challenge is to identify appropriate fairness metrics and evaluating them on validation and test data⁹⁷. By taking such proactive steps, we can ensure that the results and conclusions derived from neural engineering applications are as fair and representative as possible.

3.4 Privacy

Brain data holds a unique position in the realm of biomedical data due to its intimate connection to personal identity, thoughts, emotions, and potentially even an individual's predisposition to certain conditions or behaviors^{98,99}. With the growth of machine learning and its capacity to extract patterns from vast datasets, there is a genuine risk of deciphering personal information that a participant might not have explicitly chosen to share. As such, algorithms designed to process brain data need to include measures for protecting the privacy of individuals, including limiting the use of sensitive data, and privacy-conserving mechanisms¹⁰⁰.

3.5 Usability and Cost

While many machine learning methods are used as a step towards scientific discovery or as a preprocessing step, the final goal is often to build an implementable algorithm. The usability of

a machine learning model is multifaceted. While predictive accuracy is often a primary focus, the complexity of implementing the model can influence its practicality. As highlighted above, a seizure risk tool has been adopted both due to its interpretability but also its ease of implementation⁸¹. A model with state-of-the-art accuracy but immense computational demands might be impractical for real-time applications. Conversely, a simpler, more streamlined model might be easier to deploy and maintain in many applications. Additionally, the ease of integrating the model into existing systems, and the potential need for specialized hardware or software, can influence its usability. Implementation science, which studies the methods to promote the adoption and integration of research findings into healthcare practices, underscores the importance of these considerations, and can provide valuable insights for machine learning deployment in neural engineering¹⁰¹. Balancing the aspects of implementation complexity, computational cost, and model performance is crucial to ensure that machine learning tools are both effective and practical in the neural engineering landscape.

4. Reporting Guidelines

4.1 Existing Machine Learning Reporting Guidelines

Predictive modeling as a subfield has many guidelines for reporting results, such as the Transparent Reporting of a multivariable prediction model for Individual Prognosis or Diagnosis (TRIPOD) statement⁹ and Standards for Reporting of Diagnostic Accuracy Studies (STARD)¹⁰. These frameworks provide valuable general guidance for reporting in predictive modeling and diagnostic studies. However, they do not address the unique challenges of neural engineering, which often involve small subject numbers, complex data structures, and dependencies between data points.

The machine learning community has itself developed significant guidance on reporting. An important machine learning venue, Neural Information Processing Systems (NeurIPS), has developed and previously reported on their reproducibility program⁸. This system was largely based on the Machine Learning Reproducibility Checklist. This checklist requires authors at the time of submission to provide responses to many questions on all models and algorithms, theoretical claims, datasets, whether code to recreate experiments was included, and many details on the experimental results. After NeurIPS 2019, roughly one third of reviewers reported that they found the author checklist helpful, and reviewers gave more positive reviews to papers if they found the answers to the checklist complete and useful⁸.

While the Machine Learning Reproducibility Checklist has been instrumental in promoting reproducibility in the broader machine learning community, it lacks specific guidance on validating scientific conclusions derived from neural data. Initiatives such as the Deep Learning-EEG (DL-EEG) checklist address specific reporting needs for deep learning applied to EEG data, including algorithmic transparency and reporting consistency, but do not explicitly focus on validation process or ensuring the validity of scientific conclusions.

4.2 NERVE-ML Checklist: Purpose and Scope

Our NERVE-ML checklist is designed to fill these gaps, offering targeted guidance to ensure that machine learning methods are both reproducible and scientifically sound within the context of

neural engineering. Here we will focus more narrowly on challenges that arise in machine learning and artificial intelligence models for neural engineering. Our goal is twofold: 1) encourage reproducibility and 2) help make sure scientific and statistical conclusions are valid. As has been frequently pointed out in machine learning, a reproducible result does not ensure that it is correct. For example, including code with a paper can replicate results shown in the manuscript, but this will simply replicate any coding error. Likewise, reproducing results does not guarantee their conclusions are correct, as any flaws in methodology or validation strategy will still exist.

Hence, our checklist is focused on both the reproducibility of the machine learning methods and the scientific conclusions derived from their results. We present the NERVE-ML (Neural Engineering Reproducibility and Validity Essentials for Machine Learning) checklist, which is Shown in Table 4.

The NERVE-ML checklist is designed to ensure that key details related to data handling, machine learning methodology, and scientific conclusions are effectively conveyed. The checklist is divided into five categories, each addressing critical aspects of machine learning applications in neural engineering, to promote proper use, validation, and scientific conclusions. While the NERVE-ML checklist provides general guidance, we acknowledge that different neural data modalities, such as EEG, LFP, and fMRI, may require additional considerations. Future efforts could build on this work by developing supplementary materials tailored to these specific challenges.

4.3 Checklist Categories

4.3.1 On data and definition of training and testing set

The first category, “**On data and definition of training and testing set,**” focuses on how data is collected, processed, and split for analysis. As discussed in Section 2.2, proper data splitting is crucial to ensure that validation strategies align with the scientific question. For example, subject-level splits are necessary for generalization across individuals, while session-level splits are more appropriate for assessing temporal consistency within subjects. This category builds directly on the examples and discussions in Section 2.2 by ensuring that these considerations are documented and reported. Proper handling and reporting of these steps are critical in neural engineering, where small datasets, dependencies between samples, and potential biases can significantly affect outcomes. Explicitly addressing these factors helps prevent common errors, such as inflated performance metrics due to improper data splits.

4.3.2 On parameters and hyperparameters selection and optimization

The second category, “**On parameters and hyperparameters selection and optimization,**” addresses the choices made during model development. While our example cases in Section 2.2 do not focus specifically on hyperparameter tuning, they emphasize the importance of validation strategies, which are tightly connected to optimization procedures. This category overlaps with recommendations from prior checklists, such as the Machine Learning Reproducibility Checklist and EEG-DL. While the proposed NERVE-ML checklist takes a broader approach by also focusing on scientific validity, this overlap reinforces the importance of continuing well-established best practices and adapting them to the specific challenges of

neural engineering. By specifying whether hyperparameter tuning was conducted on independent validation sets (that are different from the final independent test set) and clearly reporting these procedures, this category ensures transparency and helps avoid overfitting or improperly tuned models, as also discussed in Section 2.6.

4.3.3 On performance metrics

The third category, **“On performance metrics,”** ensures that appropriate and interpretable evaluation metrics are reported, as discussed in Section 2.3. As also discussed in Section 2.2, validation strategies play a critical role in shaping performance estimates, making it essential to align metrics with the research question. For instance, in clinical applications, metrics like sensitivity and specificity are often necessary to assess real-world utility. This category emphasizes not only the careful selection of relevant metrics but also the importance of reporting confidence intervals or statistical significance, providing a clearer picture of the robustness and reliability of the results. This aligns with broader reporting frameworks like TRIPOD and STARD, which emphasize clear and consistent reporting of performance metrics in predictive modeling and diagnostic studies. This transparency ultimately supports better comparisons across studies and strengthens the reproducibility of machine learning in neural engineering.

4.3.4 On scientific conclusions

The fourth category, **“On scientific conclusions,”** emphasizes the critical need to link reported results to the underlying research question, ensuring that conclusions are both valid and appropriately scoped. This category highlights the importance of transparently addressing the assumptions, limitations, and generalizability of the findings, as these factors are essential for drawing scientifically rigorous conclusions in neural engineering. For example, studies employing subject-level validation can reliably infer patterns that generalize across individuals within a dataset but should avoid making claims about external generalizability without additional validation, such as multi-center data or external datasets. Similarly, claims about causal mechanisms or clinical applicability must be firmly grounded in the study design and data constraints.

In neural engineering, where datasets are often small and heterogeneous, overstated or unsupported conclusions can have significant consequences, including misleading follow-up studies or translational applications. To mitigate this risk, this category encourages authors to carefully document how their validation strategies and data structures influence the scope of their claims. This includes clearly distinguishing between exploratory and confirmatory findings and identifying any potential confounding factors that could bias the results. For instance, if confounding factors such as subject identity are not accounted for, the conclusions drawn about feature importance or model performance may be misleading.

This category also places a strong emphasis on ensuring that conclusions are actionable and meaningful within the context of neural engineering applications. For example, claims about performance improvements or clinical relevance should be supported by metrics that are aligned with the research question and target application. Additionally, researchers are encouraged to articulate how their findings could inform future research, practical applications,

or clinical implementation, while clearly acknowledging any limitations or conditions under which the results may not generalize.

The NERVE-ML checklist makes a unique contribution by addressing challenges specific to neural engineering, such as subject heterogeneity, complex data dependencies, and the need to balance reproducibility with scientific validity. While frameworks like TRIPOD and STARD emphasize transparency in linking predictive modeling results to scientific or clinical applications, NERVE-ML extends these principles to account for the nuances of neural engineering datasets and methods. By requiring researchers to explicitly connect their results to the underlying research question and application context, this category ensures conclusions are rigorously supported, transparently presented, and meaningful for advancing the field.

4.3.5 On future use of results and developed technologies

The final category, **“On future use of results and developed technologies,”** emphasizes the practical and translational implications of the study’s findings. It encourages researchers to specify the intended context of use, such as clinical applications or broader research settings, while being transparent about limitations that could impact generalizability or scalability, including small datasets or dependencies on specific devices. This category also highlights the importance of addressing ethical and societal considerations, such as patient safety, data privacy, and equity of access, to align research with responsible innovation. We note that released data and code should be in trustworthy repositories¹⁰². While we recognize that some responses in this category may appropriately be “no” due to privacy, proprietary, or other constraints, explicitly addressing these issues ensures clarity and trustworthiness. While there is some overlap with TRIPOD and STARD, this category of NERVE-ML uniquely asks researchers to document potential barriers to translating machine learning models to clinical or real-world use, such as the compatibility of preprocessing pipelines with other systems or the impact of hardware variability. By focusing on how results and technologies can be meaningfully deployed, this category ensures that findings are not only scientifically valid but also impactful and sustainable in the long term.

While this checklist includes many common items and scenarios, it is both non-exhaustive and non-universal. We acknowledge that not all items will apply to every study, and some may not be answerable due to constraints such as privacy or proprietary data. Rather than being prescriptive, this checklist is intended as a starting point to ensure that common issues are addressed and that inappropriate conclusions are avoided due to improper machine learning techniques.

Altogether, we hope this checklist will help scientists using machine learning in neural engineering to design valid and reproducible studies. Additionally, it can serve as a valuable tool for reviewers to identify potential biases and methodological flaws in neural engineering submissions, ultimately reducing the number of flawed publications in the field. By fostering rigorous reporting and validation practices, the NERVE-ML checklist directly supports our overarching goals of ensuring reproducibility and validity in machine learning research. Furthermore, by addressing the unique challenges of neural engineering, it contributes to advancing the field through transparent, trustworthy, and impactful applications of machine learning.

Finally, we note that not all the additional considerations highlighted in Section 3 are included in this checklist. While those are highly important considerations, they are often specific to the task at hand; hence we choose to focus mainly on the reproducibility and validity for the NERVE-ML checklist. We would encourage researchers to still heavily consider such issues in their work.

5. Discussion and Conclusion

Machine learning has the potential to significantly advance science, particularly in the field of neural engineering. However, this potential can only be realized if machine learning is applied with transparency, reproducibility, and scientific validity. Throughout this manuscript, we have highlighted common challenges in machine learning research, as well as issues specific to neural engineering, such as subject-specific, session-specific, and temporally smooth trends in brain data that may not reflect the outcomes of interest. As we demonstrated, the inappropriate application of validation techniques can lead to drastically inflated performance estimates, undermining the reliability of scientific conclusions.

The NERVE-ML checklist, presented in this paper, offers targeted guidelines to address these challenges by focusing on both reproducibility and validity. It builds on existing frameworks, while addressing the unique characteristics of neural engineering datasets, including small subject numbers, complex dependencies, and heterogeneous data. While initiatives like MOABB¹⁴ focus on benchmarking algorithms for brain-computer interface research, NERVE-ML takes a broader perspective by addressing reproducibility and validation across the wider range of neural engineering applications. By emphasizing practical considerations such as validation strategies, performance metrics, and the scope of scientific conclusions, the checklist ensures that machine learning research contributes meaningfully to the advancement of the field. As a next step, developing modality-specific guidelines or appendices could enhance the NERVE-ML checklist, as modalities like EEG, LFP, and fMRI present unique challenges. Future work could expand the current checklist to offer tailored guidance for these data types, ensuring reproducibility and validity across the myriad applications of machine learning in neural engineering.

In addition to guiding researchers, the NERVE-ML checklist serves as a valuable resource for reviewers, scientific publishers and funding agencies by providing a structured framework to identify potential methodological flaws and biases in submissions. By fostering rigorous and transparent practices, we hope the checklist will not only reduce the prevalence of flawed research but also contribute to a culture of robust, replicable, and impactful machine learning applications in neural engineering. Moving forward, we encourage researchers to integrate these guidelines into their work while continuing to consider broader ethical and societal implications to ensure that the field progresses in a responsible and meaningful way.

Acknowledgements

The authors would like to thank Aaron Fleming, Ph.D., and Neil Gallagher, Ph.D for feedback on the manuscript and the figures. The content is solely the responsibility of the authors and does

not necessarily represent the official views of the National Institutes of Health. F. Lotte was partially supported by projects CHIST-ERA BITSCOPE (grant ANR-21-CHRA-0003-01) and ANR PROTEUS (grant ANR-22-CE33-0015-01). R. Chavarriaga was supported by the Wellcome Trust (Grant number: 226486/Z/22/Z). B. L. Lu was supported in part by grants from STI 2030-Major Projects+2022ZD0208500, Shanghai Municipal Science and Technology Major Project (Grant No. 2021SHZDZX), Medical-Engineering Interdisciplinary Research Foundation of Shanghai Jiao Tong University "Jiao Tong Star" Program (YG2023ZD25 and YG2024ZD25), Shanghai Pilot Program for Basic Research - Shanghai Jiao Tong University (No. 21TQ1400203), and GuangCi Professorship Program of RuiJin Hospital Shanghai Jiao Tong University School of Medicine. No conflicts are reported.

Bibliography:

1. Ioannidis, J. P. A. Why Most Published Research Findings Are False. *PLoS Med* **2**, e124 (2005).
2. Roy, Y. *et al.* Deep learning-based electroencephalography analysis: a systematic review. *J Neural Eng* **16**, 051001 (2019).
3. Collins, F. S. & Tabak, L. A. Policy: NIH plans to enhance reproducibility. *Nature* **505**, 7485 505, 612–613 (2014).
4. United Nations Educational, S. and C. O. *Recommendation on the Ethics of Artificial Intelligence*. www.unesco.org/open- (2022).
5. World Health Organization. *Ethics and Governance of Artificial Intelligence for Health WHO Guidance*. (World Health Organization, Geneva, 2021).
6. Glaser, J. I., Benjamin, A. S., Farhoodi, R. & Kording, K. P. The roles of supervised machine learning in systems neuroscience. *Prog Neurobiol* **175**, 126–137 (2019).
7. Vu, M.-A. T. *et al.* A shared vision for machine learning in neuroscience. *Journal of Neuroscience* **38**, (2018).
8. Pineau, J. *et al.* Improving reproducibility in machine learning research (a report from the NeurIPS 2019 reproducibility program). *The Journal of Machine Learning Research* **22**, 1–20 (2021).
9. Collins, G. S., Reitsma, J. B., Altman, D. G. & Moons, K. G. M. Transparent reporting of a multivariable prediction model for individual prognosis or diagnosis (TRIPOD): the TRIPOD statement. *Ann Intern Med* **162**, 55–63 (2015).
10. Cohen, J. F. *et al.* STARD 2015 guidelines for reporting diagnostic accuracy studies: explanation and elaboration. *BMJ Open* **6**, e012799 (2016).
11. Collins, G. S. *et al.* Protocol for development of a reporting guideline (TRIPOD-AI) and risk of bias tool (PROBAST-AI) for diagnostic and prognostic prediction model studies based on artificial intelligence. *BMJ Open* **11**, e048008 (2021).
12. Jayaram, V. & Barachant, A. MOABB: Trustworthy algorithm benchmarking for BCIs. *J Neural Eng* **15**, (2018).
13. Sylvain Chevallier *et al.* "The largest EEG-based BCI reproducibility study for open science: the MOABB benchmark". *HAL: hal-04537061*. (2024).
14. Aristimunha, B. *et al.* Mother of all BCI Benchmarks (MOABB). Preprint at (2023).
15. Brookshire, G. *et al.* Data leakage in deep learning studies of translational EEG. *Front Neurosci* **18**, 1373515 (2024).

16. Hastie, T., Tibshirani, R. & Friedman, J. *The Elements of Statistical Learning*. Springer Series in Statistics vol. 27 (2009).
17. Deng, J. *et al.* ImageNet: A large-scale hierarchical image database. in *2009 IEEE Conference on Computer Vision and Pattern Recognition* 248–255 (IEEE, 2009). doi:10.1109/CVPR.2009.5206848.
18. Zhang, C., Bengio, S., Hardt, M., Recht, B. & Vinyals, O. Understanding deep learning (still) requires rethinking generalization. *Commun ACM* **64**, 107–115 (2021).
19. Varma, S. & Simon, R. Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics* **7**, 91 (2006).
20. Krstajic, D., Buturovic, L. J., Leahy, D. E. & Thomas, S. Cross-validation pitfalls when selecting and assessing regression and classification models. *J Cheminform* **6**, 1–15 (2014).
21. Varoquaux, G. *et al.* Assessing and tuning brain decoders: Cross-validation, caveats, and guidelines. *Neuroimage* **145**, 166–179 (2017).
22. Thomas, E., Dyson, M. & Clerc, M. An analysis of performance evaluation for motor-imagery based BCI. *J Neural Eng* **10**, 031001 (2013).
23. Varoquaux, G. Cross-validation failure: Small sample sizes lead to large error bars. *Neuroimage* **180**, 68–77 (2018).
24. Li, T.-H., Liu, W., Zheng, W. L. & Lu, B. L. Classification of Five Emotions from EEG and Eye Movement Signals: Discrimination Ability and Stability over Time. *International IEEE/EMBS Conference on Neural Engineering, NER 2019-March*, 607–610 (2019).
25. Walder-Christensen, K. *et al.* Electome network factors: Capturing emotional brain networks related to health and disease. *Cell Reports Methods* **4**, (2024).
26. Bashar, M. K., Chiaki, I. & Yoshida, H. Human identification from brain EEG signals using advanced machine learning method EEG-based biometrics. in *IECBES 2016 - IEEE-EMBS Conference on Biomedical Engineering and Sciences* 475–479 (2016). doi:10.1109/IECBES.2016.7843496.
27. Palaniappan, R. & Mandic, D. P. Biometrics from brain electrical activity: A machine learning approach. *IEEE Trans Pattern Anal Mach Intell* **29**, 738–742 (2007).
28. Gui, Q., Ruiz-Blondet, M. V., Laszlo, S. & Jin, Z. A survey on brain biometrics. *ACM Computing Surveys* vol. 51 Preprint at <https://doi.org/10.1145/3230632> (2019).
29. Deiss, O. *et al.* HAMLET: Interpretable Human And Machine co-LEarning Technique. (2018).
30. Carlson, D., Kumar, S. & Dzirasa, K. Multi-region local field potential recordings during a tail-suspension test. *Duke Research Data Repository* (2023).
31. Talbot, A., Dunson, D., Dzirasa, K. & Carlson, D. Estimating a Brain Network Predictive of Stress and Genotype with Supervised Autoencoders. *Journal of the Royal Statistical Society: Series C* (2023).
32. Carlson, D. *et al.* Dynamically Timed Stimulation of Corticolimbic Circuitry Activates a Stress-Compensatory Pathway. *Biol Psychiatry* **82**, (2017).
33. van Enkhuizen, J., Minassian, A. & Young, J. W. Further evidence for Clock-Δ19 mice as a model for bipolar disorder mania using cross-species tests of exploration and sensorimotor gating. *Behavioural Brain Research* **249**, 44–54 (2013).

34. Abelaira, H. M., Reus, G. Z. & Quevedo, J. Animal models as tools to study the pathophysiology of depression. *Revista Brasileira de Psiquiatria* **35**, S112--S120 (2013).
35. Nakanishi, M. *et al.* Questionable Classification Accuracy Reported in 'Designing a Sum of Squared Correlations Framework for Enhancing SSVEP-Based BCIs'. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **28**, 1042–1043 (2020).
36. Zhang, C., Bengio, S., Hardt, M., Recht, B. & Vinyals, O. Understanding deep learning requires rethinking generalization. (2016).
37. Bates, S., Hastie, T. & Tibshirani, R. Cross-Validation: What Does It Estimate and How Well Does It Do It? *J Am Stat Assoc* **2024**, 1434–1445 (2024).
38. Liu, W., Qiu, J. L., Zheng, W. L. & Lu, B. L. Comparing Recognition Performance and Robustness of Multimodal Deep Learning Models for Multimodal Emotion Recognition. *IEEE Trans Cogn Dev Syst* **14**, 715–729 (2022).
39. Zhao, L. M., Li, R., Zheng, W. L. & Lu, B. L. Classification of Five Emotions from EEG and Eye Movement Signals: Complementary Representation Properties. *International IEEE/EMBS Conference on Neural Engineering, NER* **2019-March**, 611–614 (2019).
40. Benomar, M., Cao, S., Vishwanath, M., Vo, K. & Cao, H. Investigation of EEG-Based Biometric Identification Using State-of-the-Art Neural Architectures on a Real-Time Raspberry Pi-Based System. *Sensors* **2022, Vol. 22, Page 9547** **22**, 9547 (2022).
41. Hema, C. R. & Osman, A. A. Single trial analysis on EEG signatures to identify individuals. *IEEE* 1–3 (2010) doi:10.1109/CSPA.2010.5545313.
42. Kemp, B., Zwinderman, A. H., Tuk, B., Kamphuisen, H. A. C. & Oberyé, J. J. L. Analysis of a sleep-dependent neuronal feedback loop: The slow-wave microcontinuity of the EEG. *IEEE Trans Biomed Eng* **47**, 1185–1194 (2000).
43. Van Der Donckt, J. *et al.* Do not sleep on traditional machine learning: Simple and interpretable techniques are competitive to deep learning for sleep scoring. *Biomed Signal Process Control* **81**, 104429 (2023).
44. Supratak, A., Dong, H., Wu, C. & Guo, Y. DeepSleepNet: A model for automatic sleep stage scoring based on raw single-channel EEG. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **25**, 1998–2008 (2017).
45. Supratak, A. & Guo, Y. TinySleepNet: An Efficient Deep Learning Model for Sleep Stage Scoring based on Raw Single-Channel EEG. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS* **2020-July**, 641–644 (2020).
46. Perslev, M. *et al.* U-Sleep: resilient high-frequency sleep staging. *npj Digital Medicine* **2021 4:1** **4**, 1–12 (2021).
47. Phan, H. *et al.* SleepTransformer: Automatic Sleep Staging With Interpretability and Uncertainty Quantification. *IEEE Trans Biomed Eng* **69**, 2456–2467 (2022).
48. Phan, H. *et al.* XSleepNet: Multi-View Sequential Model for Automatic Sleep Staging. *IEEE Trans Pattern Anal Mach Intell* **44**, 5903–5915 (2022).
49. Guillot, A. & Thorey, V. RobustSleepNet: Transfer Learning for Automated Sleep Staging at Scale. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **29**, 1441–1451 (2021).

50. Parks, D. F. *et al.* A nonoscillatory, millisecond-scale embedding of brain state provides insight into behavior. *Nature Neuroscience* 2024 1–15 (2024) doi:10.1038/s41593-024-01715-2.
51. Carson, W. E. *et al.* Model selection to achieve reproducible associations between resting state EEG features and autism. *Scientific Reports* 2024 14:1 **14**, 1–17 (2024).
52. Halevy, A., Norvig, P. & Pereira, F. The Unreasonable Effectiveness of Data. *IEEE Intell Syst* **24**, 8–12 (2009).
53. Vabalas, A., Gowen, E., Poliakoff, E. & Casson, A. J. Machine learning algorithm validation with a limited sample size. *PLoS One* **14**, e0224365 (2019).
54. Golland, P. & Fischl, B. Permutation Tests for Classification: Towards Statistical Significance in Image-Based Studies. in *Biennial International Conference on Information Processing in Medical Imaging* 330–341 (Springer, Berlin, Heidelberg, 2003). doi:10.1007/978-3-540-45087-0_28.
55. Combrisson, E. & Jerbi, K. Exceeding chance level by chance: The caveat of theoretical chance levels in brain signal classification and statistical assessment of decoding accuracy. *J Neurosci Methods* **250**, 126–136 (2015).
56. Pencina, M. J., D'Agostino, R. B., D'Agostino, R. B. & Vasan, R. S. Evaluating the added predictive ability of a new marker: From area under the ROC curve to reclassification and beyond. *Stat Med* **27**, 157–172 (2008).
57. Siontis, G. C. M., Tzoulaki, I., Castaldi, P. J. & Ioannidis, J. P. A. External validation of new risk prediction models is infrequent and reveals worse prognostic discrimination. *J Clin Epidemiol* **68**, 25–34 (2015).
58. Labs, C. at R., Sussillo, D., Kaifosh, P. & Reardon, T. A generic noninvasive neuromotor interface for human-computer interaction. *bioRxiv* 2024.02.23.581779 (2024) doi:10.1101/2024.02.23.581779.
59. Jacot, A., Gabriel, F. & Hongler, C. Neural Tangent Kernel: Convergence and Generalization in Neural Networks. *Adv Neural Inf Process Syst* **2018-Decem**, 8571–8580 (2018).
60. Belkin, M., Hsu, D., Ma, S. & Mandal, S. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences* **116**, 15849–15854 (2019).
61. Ganin, Y. *et al.* Domain-adversarial training of neural networks. *Journal of Machine Learning Research* **17**, 1–35 (2016).
62. Li, Y., Murias, M., Major, S., Dawson, G. & Carlson, D. E. Extracting Relationships by Multi-Domain Matching. *Neural Information Processing Systems* (2018).
63. Ash, J. T., Schapire, R. E. & Engelhardt, B. E. Unsupervised Domain Adaptation Using Approximate Label Matching. *arXiv preprint arXiv:1602.04889* (2016).
64. Li, D., Yang, Y., Song, Y.-Z. & Hospedales, T. M. Learning to Generalize: Meta-Learning for Domain Generalization. *Thirty-Second AAAI Conference on Artificial Intelligence* (2018).
65. Zoph, B. & Le Google Brain, Q. V. Neural Architecture Search with Reinforcement Learning. *International Conference on Learning Representations* Preprint at (2017).
66. Shahriari, B., Swersky, K., Wang, Z., Adams, R. P. & de Freitas, N. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE* **104**, (2015).
67. Dominguez, L. G. On the risk of extracting relevant information from random data. *J Neural Eng* **6**, 058001 (2009).

68. Chau, T. & Damouras, S. Reply to 'On the risk of extracting relevant information from random data'. *J Neural Eng* **6**, 058002 (2009).
69. Yuan, Z., Zhang, D., YANG, Y., Chen, J. & Li, Y. PPI: Pretraining Brain Signal Model for Patient-independent Seizure Detection. *Adv Neural Inf Process Syst* **36**, 69586–69604 (2023).
70. Liu, R. *et al.* Drop, Swap, and Generate: A Self-Supervised Approach for Generating Neural Activity. *Adv Neural Inf Process Syst* **34**, 10587–10599 (2021).
71. Schneider, S., Lee, J. H. & Mathis, M. W. Learnable latent embeddings for joint behavioural and neural analysis. *Nature* **2023 617:7960** **617**, 360–368 (2023).
72. Pereira, T. D. *et al.* SLEAP: A deep learning system for multi-animal pose tracking. *Nature Methods* **2022 19:4** **19**, 486–495 (2022).
73. Dunn, T. W. *et al.* Geometric deep learning enables 3D kinematic profiling across species and environments. *Nature Methods* **2021 18:5** **18**, 564–573 (2021).
74. Jiang, W. B., Zhao, L. M. & Lu, B. L. Large Brain Model for Learning Generic Representations with Tremendous EEG Data in BCI. *12th International Conference on Learning Representations, ICLR 2024* (2024).
75. Tu, T. *et al.* Towards Generalist Biomedical AI. *NEJM AI* **1**, (2024).
76. Rudin, C. & Carlson, D. The Secrets of Machine Learning: Ten Things You Wish You Had Known Earlier to be More Effective at Data Analysis. *Operations Research & Management Science in the Age of Analytics* 44–72 (2019) doi:10.1287/educ.2019.0200.
77. Rudin, C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence* **2019 1:5** **1**, 206–215 (2019).
78. Rudin, C. Why black box machine learning should be avoided for high-stakes decisions, in brief. *Nature Reviews Methods Primers* **2022 2:1** **2**, 1–2 (2022).
79. Fellous, J.-M., Sapiro, G., Rossi, A., Mayberg, H. & Ferrante, M. Explainable Artificial Intelligence for Neuroscience: Behavioral Neurostimulation. *Front Neurosci* **13**, 1346 (2019).
80. Nori, H., Jenkins, S., Koch, P. & Caruana, R. InterpretML: A Unified Framework for Machine Learning Interpretability. (2019).
81. Struck, A. F. *et al.* Association of an electroencephalography-based risk score with seizure probability in hospitalized patients. *JAMA Neurol* **74**, 1419–1424 (2017).
82. Kaur, H. *et al.* Interpreting Interpretability: Understanding Data Scientists' Use of Interpretability Tools for Machine Learning. *Conference on Human Factors in Computing Systems - Proceedings* (2020) doi:10.1145/3313831.3376219.
83. Gallagher, N., Dzirasa, K. & Carlson, D. Directed Spectrum Measures Improve Latent Network Models Of Neural Populations. *Neural Information Processing Systems* <https://proceedings.neurips.cc/paper/2021/hash/3d36c07721a0a5a96436d6c536a132ec-Abstract.html> (2021).
84. Alagapan, S. *et al.* Cingulate dynamics track depression recovery with deep brain stimulation. *Nature* **2023 622:7981** **622**, 130–138 (2023).
85. Hultman, R. *et al.* Brain-wide Electrical Spatiotemporal Dynamics Encode Depression Vulnerability. *Cell* **173**, 166–180.e14 (2018).

86. Mague, S. D. *et al.* Brain-wide electrical dynamics encode individual appetitive social behavior. *Neuron* **110**, 1728–1741.e7 (2022).
87. Tozzi, L. *et al.* Personalized brain circuit scores identify clinically distinct biotypes in depression and anxiety. *Nature Medicine* **2024 30:7 30**, 2076–2087 (2024).
88. Trocellier, D., N’Kaoua, B. & Lotte, F. Visual cues can bias EEG Deep Learning models. *Neuroergonomics* (2024).
89. Dhar, P. The carbon impact of artificial intelligence. *Nat Mach Intell* **2**, 423–425 (2020).
90. Grossman, Y. S. *et al.* Brain-wide oscillatory network selectively encodes aggression. *bioRxiv* (2022) doi:10.1101/2022.12.07.519272.
91. Lotte, F. *et al.* A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update. *J Neural Eng* **15**, 031005 (2018).
92. Lotte, F. Signal processing approaches to minimize or suppress calibration time in oscillatory activity-based brain-computer interfaces. *Proceedings of the IEEE* **103**, 871–890 (2015).
93. Caliskan, A., Bryson, J. J. & Narayanan, A. Semantics derived automatically from language corpora contain human-like biases. *Science* (1979) **356**, 183–186 (2017).
94. Webb, E. K., Etter, J. A. & Kwasa, J. A. Addressing racial and phenotypic bias in human neuroscience methods. *Nature Neuroscience* **2022 25:4 25**, 410–414 (2022).
95. Wassenaar, E. B. & Van den Brand, J. G. H. Reliability of near-infrared spectroscopy in people with dark skin pigmentation. *J Clin Monit Comput* **19**, 195–199 (2005).
96. Caton, S. & Haas, C. Fairness in Machine Learning: A Survey. *ACM Comput Surv* (2023) doi:10.1145/3616865.
97. Castelnovo, A. *et al.* A clarification of the nuances in the fairness metrics landscape. *Scientific Reports* **2022 12:1 12**, 1–21 (2022).
98. Farahany, N. A. The battle for your brain : defending the right to think freely in the age of neurotechnology. 277.
99. Ramos, K. M. *et al.* NeuroView The NIH BRAIN Initiative: Integrating Neuroethics and Neuroscience. (2019) doi:10.1016/j.neuron.2019.01.024.
100. Dwork, C. & Roth, A. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends? in Theoretical Computer Science* **9**, 211–407 (2013).
101. Chambers, D. A., Feero, W. G. & Khoury, M. J. Convergence of Implementation Science, Precision Medicine, and the Learning Health Care System: A New Model for Biomedical Research. *JAMA* **315**, 1941–1942 (2016).
102. Lin, D. *et al.* The TRUST Principles for digital repositories. *Scientific Data* **2020 7:1 7**, 1–5 (2020).

Table 1: Results on multiple prediction tasks on the LFP dataset using a multi-layer perceptron. Parentheses denote standard deviation of the cross-validation splits. The table reports predictive performance under different approaches, including reporting on training data, a 5-Fold Cross-Validation over Instances, and 5-Fold Cross Validation over Subjects.

Target / Metric	Stress / AUC	Genotype / AUC	Subject ID / Acc.	Random / AUC
No Validation	1.000 (N/A)	1.000 (N/A)	0.998 (N/A)	1.000 (N/A)
5-Fold over Instances	0.932 (0.002)	0.980 (0.001)	0.949 (0.001)	0.979 (0.005)
5-Fold over Subjects	0.886 (0.014)	0.639 (0.118)	N/A	0.579 (0.129)

Table 2: Results on multiple prediction tasks on the SEED-V dataset using a multi-layer perceptron. Parentheses denote standard deviation of the cross-validation splits. To align our results with those of previous studies, we employ a 3-fold cross-validation by subject. Specifically, for each subject, the 15 film clips in each session are divided into three segments: the first 5 clips, the middle 5 clips, and the last 5 clips. Data from the three sessions are then concatenated. Models are trained on permutations of two of the three segments and tested on the remaining third segment to evaluate performance. $K=5$ for all experiments except for session identity, which uses $K=3$.

Target / Metric	Emotion Classification / Acc.	Subject ID / Acc.	Session ID / Acc.	Random by Subject / Acc.	Random by Session / Acc.
No Validation	1.000 (N/A)	1.000 (N/A)	1.000 (N/A)	1.000 (N/A)	1.000 (N/A)
K-Fold	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	1.000 (.000)
3-fold by subject	0.719 (0.096)	N/A	N/A	N/A	N/A
Subject/Group K-Fold	0.318 (0.048)	N/A	0.397 (0.050)	0.209 (0.156)	0.384 (0.084)
Session/Group K-Fold	0.481 (0.029)	0.972 (0.011)	N/A	0.937 (0.027)	N/A
Trial/Group K-Fold	0.494 (0.089)	0.997 (0.004)	0.997 (0.004)	0.996 (0.006)	0.997 (0.004)

Table 3: Results on multiple prediction tasks on the SC-EDF-20 dataset using a multi-layer perceptron. Parentheses denote standard deviation of the cross-validation splits. K=5 in all experiments.

Target / Metric	Sleep Stage / Acc.	Subject ID / Acc.	Subject-Random / Acc.	Subject-Permutation / Acc.
No Validation	1.000 (N/A)	1.000 (N/A)	1.000 (N/A)	1.000 (N/A)
K-Fold	0.891 (0.003)	0.966 (.0120)	0.969 (0.004)	0.588 (0.017)
Subject/Group K-Fold	0.827 (0.029)	N/A	0.141 (0.096)	0.323 (0.044)
Time K-Fold	0.831 (0.013)	0.772 (0.060)	0.838 (0.062)	0.383 (0.012)

Table 4: NERVE-ML (Neural Engineering Reproducibility and Validity Essentials for Machine Learning) Checklist

Question	Yes	No	Unclear	N/A
On data and definition of training and testing set:				
Is there appropriate detail on all data sources (e.g., origin of the data, relevant summary statistics, preprocessing, etc.)?				
Training and testing data should be statistically independent, such that they are not only different data but also not dependent on each other. In particular:				
For data based on epochs/time windows of neural signals (as typical in neural engineering applications), training epochs and testing epochs do not overlap and are not consecutive to each other.				
If the data acquisition protocol has a block structure (e.g., same class label for all data points of that block), then the test data comes from different blocks than the training data.				
On parameters and hyperparameters selection and optimization:				
Are the assumptions of the machine learning approaches or models clearly defined and documented?				
In general, the parameters and settings/hyperparameters should be selected on training / validation data that are completely independent of the testing data. In particular:				
Feature selection was performed using training data only (i.e., not using testing data and not using all data)				
Machine learning settings/hyperparameters are optimized only on training data. This should be the case not only for a classifier, but also for any algorithm whose settings are optimized on data.				
Any normalization parameters (e.g. data mean and standard deviation) was estimated on training data only. Testing data should not be included in this estimation (e.g., this estimation should not be made on all available data)				
Hyperparameters (e.g., regularization parameters of SVM or Logistic regression, learning rate, batch size, etc.) were selected on training and/or validation data only.				
The architecture of a neural network was chosen on training/validation data only.				
On performance metrics:				
Are the chosen performance metrics clearly defined and motivated?				
For unbalanced test data (e.g., a significantly different amount of testing data per class), was a performance metric that can handle unbalanced data used (e.g., Area Under the ROC Curve or balanced accuracy)?				

For a test set with little data, was the chance level performance metric reported (either through analytical calculations or permutation testing) and compared to the actual performance?				
Are measures of uncertainty clearly defined and appropriate?				
On scientific conclusions:				
Are the key scientific questions clearly related to model results?				
Is it clear how the validation technique relates to the scientific questions?				
Is the practical significance of the performance metrics clearly stated?				
Are statistics run on model results?				
Are statistical assumptions clearly defined?				
Is it clear what level of generalization (e.g., inter-subject, inter-cohort, inter-genotype, etc.) is tested?				
Are the limitations of the results clearly stated?				
Are alternative explanations for performance considered?				
Are there concerns about bias or fairness?				
Were specific metrics on bias and fairness identified and evaluated?				
On future use of results and developed technologies:				
Will the data be publicly available?				
Will data include necessary metadata for reproducing results?				
Has the data been properly anonymized according to data privacy protection laws and good practices?				
Does anonymization of the data prevent reproduction of the results?				
Is the data stored in a trustworthy repository?				
Is the data license stated?				
Will the code be released in full, including training code, evaluation code, and visualizations?				
Is the code license stated?				
Are dependencies and software versions included?				
Is it stated which results can be fully replicated by the code?				
Is there an appropriately detailed guide on how to use the code?				
Will the trained machine learning model be released?				

Is it clear what resources are necessary to run the code (computational resources, software environment, computational complexity)?				
Is it stated how long any released material will be available?				
It is stated whether permanent record will be created, typically a Digital Object Identifier (DOI)?				

Figure 1: **Overview of a typical machine learning pipeline.** (a) All subjects are randomly split into train, validation, and test data as described in the text. Blue, red, and green denote the training, validation, and test process, respectively. (b) All training subjects have training data, which may comprise of either a single datum or multiple instances for each subject along with known targets in supervised machine learning. For each possible algorithm setting (e.g., type and shape of neural network, penalization strength, etc.), the machine learning method (or algorithm) is trained or learned from the training data and targets. After the algorithm has been learned, it is evaluated on the validation data to estimate performance on data unseen by the training approach. (c) The best performing algorithm is chosen based on estimated performance on the validation data. As that algorithm is chosen based on the best validation performance, it is a biased estimate due to the many comparisons. Thus, for that single chosen algorithm, it is evaluated on the test data to give a more reliable estimate of performance.

Figure 2: **Visualization of Cross-Validation Procedure.** On the left, we show 6 subjects, each labeled with a subject ID ($S_1, S_2, \dots$). For a 3-fold cross validation, there are 3 different groups of data, and each subject is assigned to a single group. For each fold, 2 of groups are used as training data, and the remaining group is used as validation data. The training data is used to train the machine learning method, and the validation data is used to estimate performance. The performance is averaged over all folds to get a more reliable performance estimate for the chosen algorithm settings.

Figure 3: **Visualization of Data Structure.** Here, we visualize a single individual that has data collected over multiple sessions. For simplicity, we show a 3-channel temporal signal such as would be collected by an electrophysiological experiment, but many different types of data could be used instead. Each session has a separate start and stop point and is split into multiple instances or epochs and associated with a target. On the right, we show a representation of a single individual over the sessions, which is used to describe splitting strategies in Figure 4.

Figure 4: **Visualization of different types of data splitting.** In an experiment where there are repeated samples from subjects, the data splitting changes the statistical question being asked and estimated performance. 3 different types of cross-validation procedures are shown on 3 subjects for visual clarity, each marked with a subject ID (S_1, S_2, S_3). In practice, this process

would happen on many more subjects. (a) *Split by Instance*. The data is split so that each individual data instance is split randomly into the training (blue) or test/validation (red) data. Each subject will have some data in the training and test data without regards to the session structure. This is visualized with 2 folds for visual clarity but can be done with any number of folds. (b) *Split by Session*. Here, each subject is split so that each session is in a different cross-validation fold. Each subject has some data in each fold, but only with regards to each session. The training may proceed on either each subject, or on all subjects simultaneously depending on the statistical question. (c) *Split by Subject*. Each subject's data is completely in a single cross-validation fold, so a subject is either completely in a training or validation data for each training loop.

Figure 5: **Visualization of subject-level randomization procedure.** (middle) Each subject is assigned a random label or outcome, either by shuffling subject-level labels (permutation testing) or by drawing the outcome from a random number generator. (left) In this procedure, each instance of data in a subject is assigned the same label. (right) Using different cross validation procedures results in drastically different performance. Because of the randomization procedure, the optimal population-level accuracy is 50% assuming balanced classes. Splitting by instance yields near perfect predictions as the machine learning system learns to identify individuals rather than identifying information related to the outcome of interest. Splitting by subject yields performance estimates indistinguishable from chance (50%). See Table 1 for numerical results on an example case.

Figure 6: **Visualization of time structure and cross-subject permutations.** (a) Visualization of a validation split over time, where each subject is split into 5 sequential chunks even though data can be thought of coming from a single session. The top shows Subject 1 (S1)'s data in the first fold and third fold of the cross-validation process, showing the evolution of the test set. (b) Visualization of the random permutation between subjects. The top shows a hypothetical relationship between data and the target sleep state (here Wake (W) and Rapid Eye Movement (REM)) as measured. The complete sequence of targets is copied and matched with data from a random subject, which breaks the relationship between the data sequence and the sleep state but keeps the temporal structure.

Appendix:

Glossary of Key Terms

- **Training Set:** A subset of the dataset used to train the model. The training set is utilized to learn the model's parameters, aiming to minimize errors in predictions during the training process.
- **Validation Set:** A subset of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyperparameters. The validation set is used repeatedly to estimate the performance of multiple models or model configurations.
- **Test Set:** A subset of data used only once to assess the performance of a fully trained model, providing an unbiased evaluation of the final model's performance on new, previously unseen data.
- **Cross-Validation:** A statistical method used to evaluate performance of machine learning models. It is commonly used when the dataset is limited in size. The data set is divided into k subsets (folds), and the holdout method is repeated k times. Each time, one of the k subsets (folds) is used as the test set and the other $k-1$ subsets (folds) are put together to form a training set.
- **Generalization Error:** The measures of how accurately a machine learning model performs on new, unseen data. It reflects the ability of the model to adapt to the variability in new data beyond the training set.
- **Hyperparameters:** Parameters that define the model architecture and how the model is learned, including learning rate, number of layers, number of nodes in each layer, etc. These parameters are set before training the model.
- **Overfitting:** A modeling error in machine learning which occurs when a function is too closely fitted to a limited set of data points. It usually results in a model that performs well on training data but poorly on unseen data due to its complex structure.
- **Confidence Interval:** An interval that is likely to contain the true population parameter (e.g., model performance) with a certain level of confidence. Confidence intervals provide a measure of the uncertainty in performance estimates.

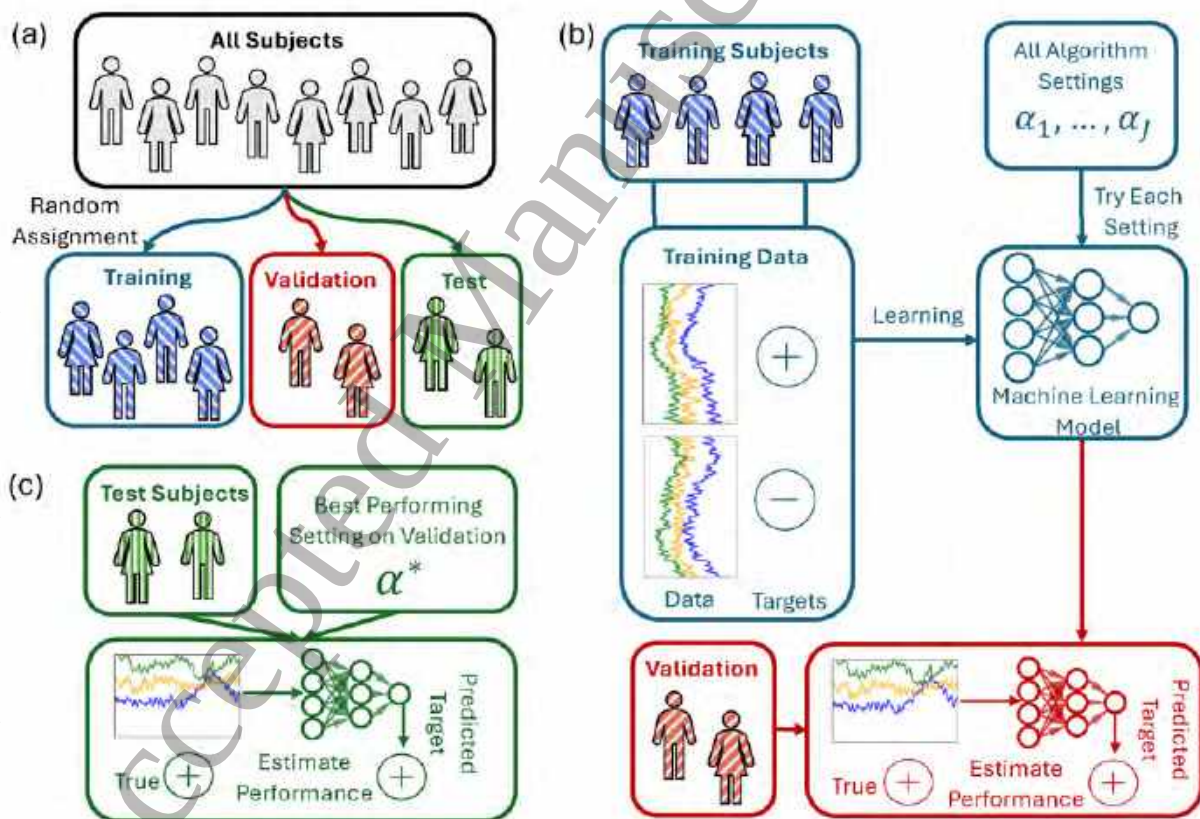


Figure 1

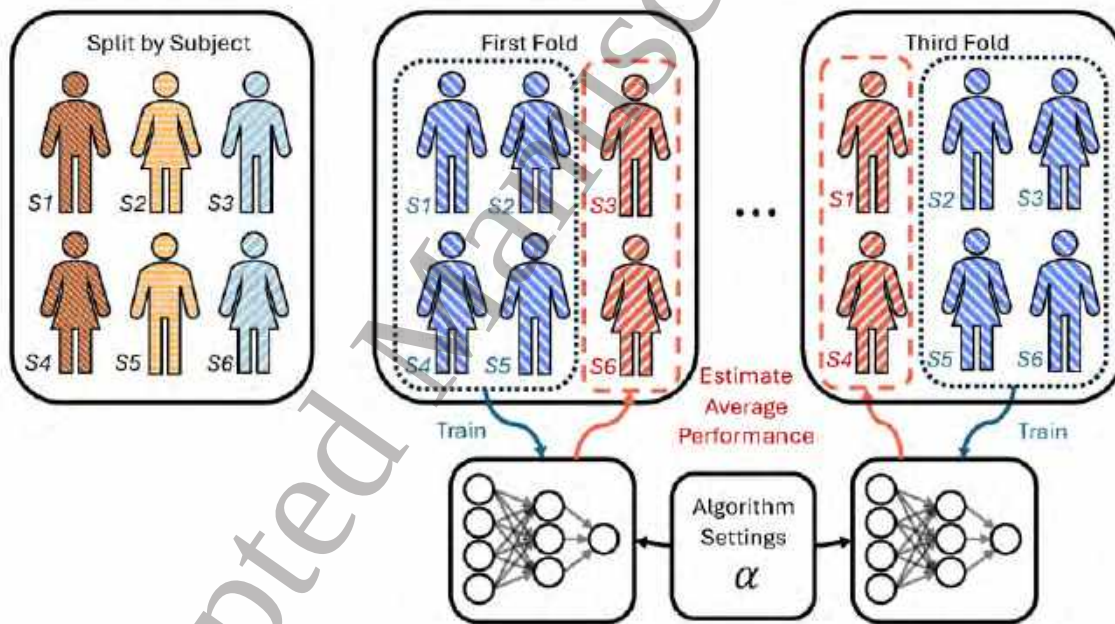


Figure 2

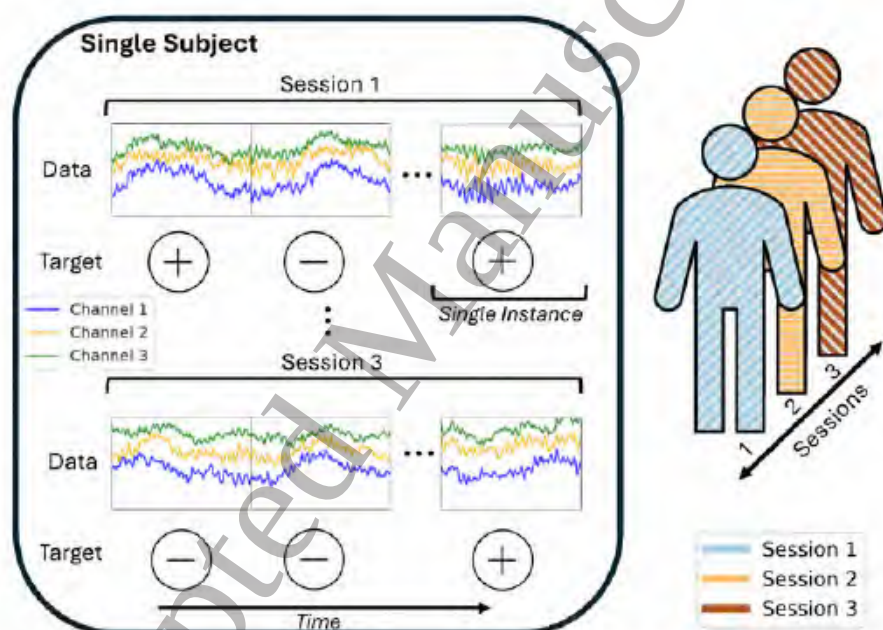


Figure 3

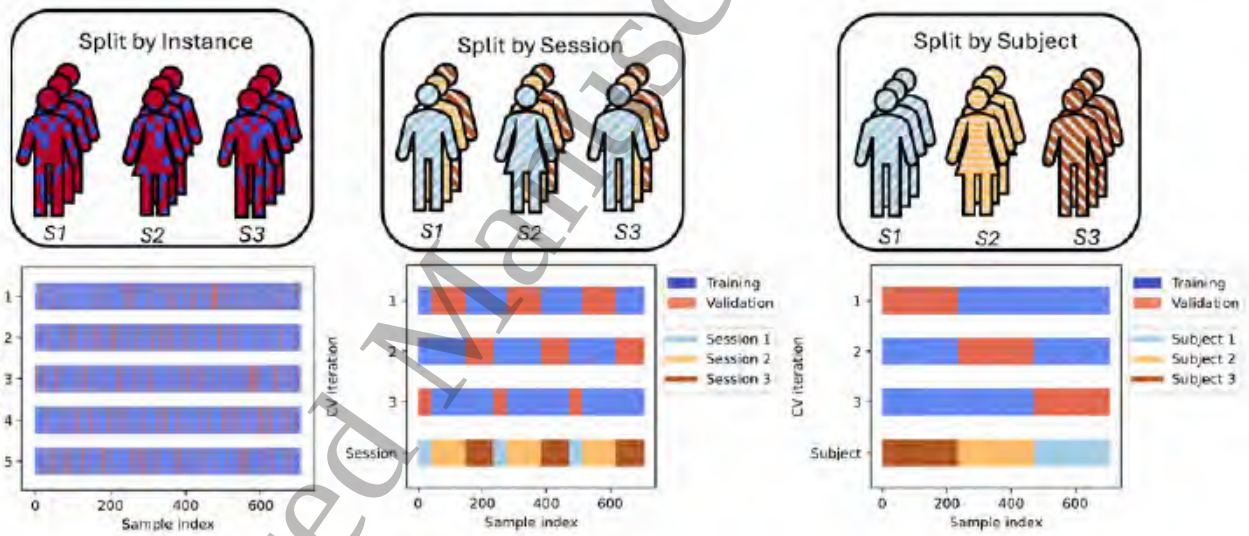


Figure 4

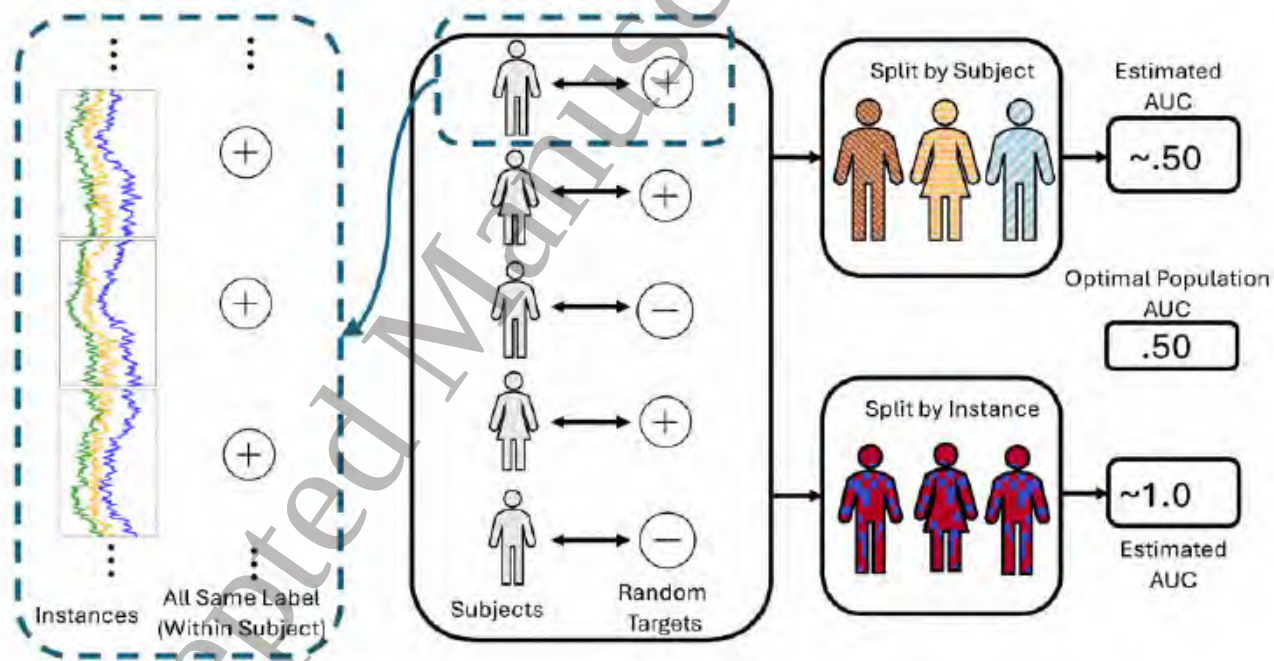


Figure 5

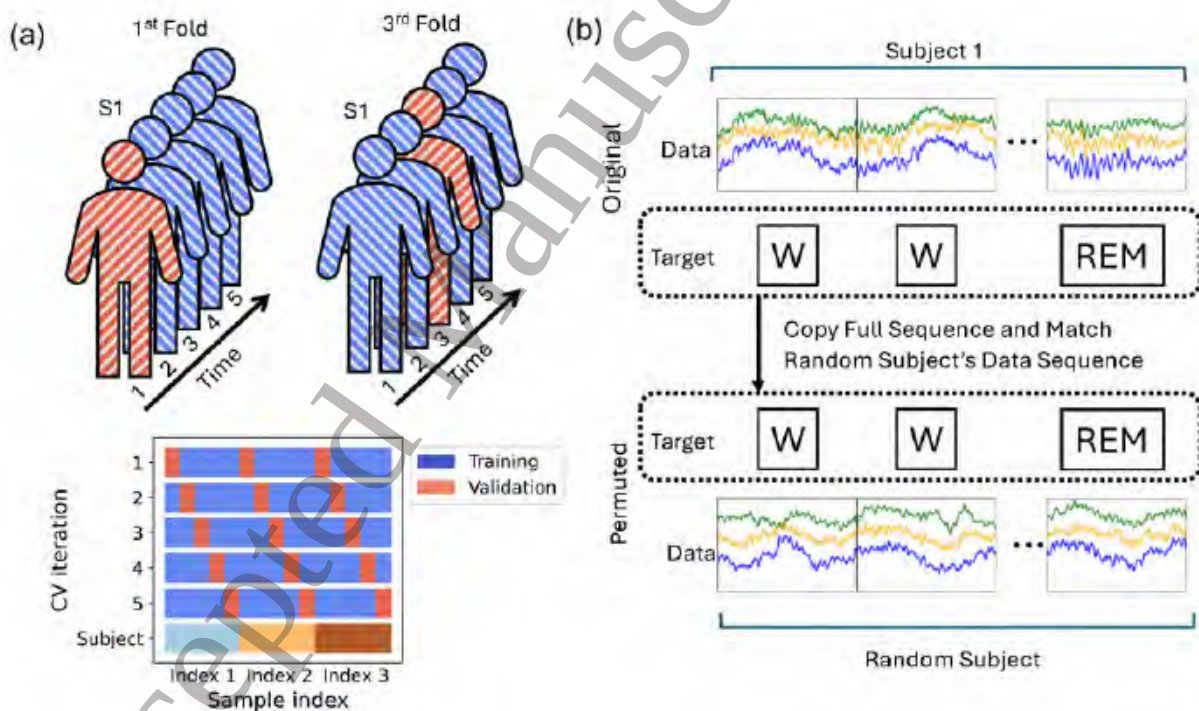


Figure 6

Appendix

A.1 List of Publications – Volume 2

Machine Perception and Cognition Group

P. Yan, A. Abdulkadir, P.-P. Luley, M. Rosenthal, G. A. Schatte, B. F. Grewe, T. Stadelmann. *A comprehensive survey of deep transfer learning for anomaly detection in industrial time series : methods, applications, and directions.* January, 2024, DOI: 10.1109/ACCESS.2023.3349132.

L. Tuggener, R. Emberger, A. Ghosh, P. Sager, Y. P. Satyawan, J. Montoya, S. Goldschagg, F. Seibold, U. Gut, P. Ackermann, J. Schmidhuber, T. Stadelmann. *Real world music object recognition. Transactions of the International Society for Music Information Retrieval.* January, 2024, DOI: 10.5334/tismir.157.

P. Yan, A. Abdulkadir, G. Aguzzi, G. A. Schatte, B. F. Grewe, T. Stadelmann. *Automated process monitoring in injection molding via representation learning and setpoint regression.* May, 2024, DOI: 10.1109/SDS60720.2024.00027.

L. Tuggener, P. Sager, Y. Taoudi-Benchekroun, B. F. Grewe, T. Stadelmann. *So you want your private LLM at home? : a survey and benchmark of methods for efficient GPTs.* May, 2024, DOI: 10.1109/SDS60720.2024.00036.

F. Schmitt-Koopmann, E. M. Huang, H.-P. Hutter, T. Stadelmann, A. Darvishy. *MathNet : a data-centric approach for printed mathematical expression recognition.* May, 2024, DOI: 10.1109/ACCESS.2024.3404834.

B. Meyer, T. Stadelmann, M. Lüthi. *ScalaGrad : a statically typed automatic differentiation library for safer data science.* May, 2024, DOI: 10.1109/SDS60720.2024.00040.

D. Neururer, V. Dellwo, T. Stadelmann. *Deep neural networks for automatic speaker recognition do not learn supra-segmental temporal features.* May, 2024, DOI: 10.1016/j.patrec.2024.03.016.

A. Dashti, T. Stadelmann, T. Kohl. *Machine learning for robust structural uncertainty quantification in fractured reservoirs.* June, 2024, DOI: 10.1016/j.geothermics.2024.103012.

P. R. Jermain, M. Oswald, T. Langdun, S. Wright, A. Khan, T. Stadelmann, A. Abdulkadir, A. N. Yaroslavsky. *Deep learning-based cell segmentation for rapid optical cytopathology of thyroid cancer.* July, 2024, DOI: 10.1038/s41598-024-64855-2.

A. Begga, W. Ali, G. Niculescu, F. Escolano, T. Stadelmann, M. Pelillo. *Community-hop: enhancing node classification through community preference.* September, 2024, DOI: 10.21256/zhaw-31438.

W. Ali, S. Vascon, T. Stadelmann, M. Pelillo. *Hierarchical glocal attention pooling for graph classification.* September, 2024, DOI: 10.1016/j.patrec.2024.09.009.

P. Bolt, V. Ziebart, C. Jaeger, N. Schmid, **T. Stadelmann**, **R. M. Fuchsli**. *A simulation study on energy optimization in building control with reinforcement learning*. October, 2024, DOI: 10.1007/978-3-031-71602-7_27.

T. Stadelmann. *A guide to AI : understanding the technology, applying it successfully, and shaping a positive future*. January, 2025, DOI: 10.21256/zhaw-32180.

Natural Language Processing Group

P. von Däniken, J. M. Deriu, A. Rodrigo, M. Cieliebak. *Improving Quantification with Minimal In-Domain Annotations: Beyond Classify and Count.* June, 2024, DOI: 10.1609/icwsm.v18i1.31411.

J. Michot, M. Hürlimann, J. Deriu, L. Sauer, K. Mlynchyk, M. Cieliebak. *Error-preserving Automatic Speech Recognition of Young English Learners' Language.* August, 2024, DOI: 10.18653/v1/2024.acl-long.348.

P. von Däniken, J. Deriu, D. Tuggener, M. Cieliebak. *Favi-Score: A Measure for Favoritism in Automated Preference Ratings for Generative AI Evaluation.* June, 2024, DOI: arXiv:2406.01131, <https://arxiv.org/abs/2406.01131>.

D. Tuggener, T. Schneider, A. Huwiler, T. Kreienbühl, S. Hischier, P. von Däniken, S. Niehaus. *Role-Playing LLMs in Professional Communication Training: The Case of Investigative Interviews with Children.* September, 2024. DOI: 10.21256/zhaw-31482.

Responsible AI Innovation Group

F. Iwane, T. Porssut, O. Blanke, **R. Chavarriaga**, J. d. R. Millán, B. Herbelin, R. Boulic. *Customizing the human-avatar mapping based on EEG error related potentials during avatar-based interaction*. March, 2024, DOI: 10.1088/1741-2552/ad2c02.

Y. Billeter, P. Denzel, R. Chavarriaga, O. Forster, F.-P. Schilling, S. Brunner, C. Frischknecht-Gruber, M. Reif, **J. Weng**. *MLOps as enabler of trustworthy AI*. May, 2024, DOI: 10.1109/SDS60720.2024.00013.

P. Denzel, S. Brunner, **Y. Billeter, O. Forster**, C. Frischknecht-Gruber, M. Reif, **F.-P. Schilling, J. Weng, R. Chavarriaga**, A. Amini, M. Repetto, A. Iranfar. *Towards the certification of AI-based systems*. May, 2024, DOI: 10.1109/SDS60720.2024.00020.

Y. Billeter, S. Brunner, **R. Chavarriaga, P. Denzel, O. Forster**, C. M.-L. Frischknecht-Gruber, M. Reif, **F.-P. Schilling, J. Weng**, A. Iranfar, M. Repetto. *A Certification Scheme For Artificial Intelligence Based Systems*. June, 2024. DOI: 10.21256/zhaw-30549.

O. Lone, **M. Weiss**, D. Baumgartner, **R. Chavarriaga**. *Python-based open Software for EEG-based Brain-Machine Interfaces*. September 2024, DOI: 10.21256/zhaw-31394.

M. Sultana, O. Jain, S. Halder, A. Matran-Fernandez, R. Nawaz, R. Scherer, **R. Chavarriaga**, J. d. R. Millán, S. Perdakis. *Evaluating dry EEG technology out of the lab*. October, 2024, DOI: 10.1109/metroxraine62247.2024.10797021.

P. Denzel, O. Forster, Y. Billeter, F.-P. Schilling, R. Chavarriaga, C. M.-L. Frischknecht-Gruber, S. Brunner, M. Reif, **J. Weng**. *certAInty – A Certification Scheme for AI Trustworthiness An Innosuisse Project*. January, 2025. DOI: 10.21256/zhaw-32422.

C. Frischknecht-Gruber, **P. Denzel**, M. Reif, **Y. Billeter**, S. Brunner, **O. Forster, F.-P. Schilling, J. Weng, R. Chavarriaga**. *AI assessment in practice : implementing a certification scheme for AI trustworthiness*. January 2025, DOI: 10.4230/OASlcs.SAIA.2024.15.

G. Starke, F. Gille, A. Termine, Y. S. J. Aquino, **R. Chavarriaga**, A. Ferrario, J. Hastings, K. Jongsma, P. Kellmeyer, B. Kulynych, E. Postan, E. Racine, D. Sahin, P. Tomaszewska, K. Vold, J. Webb, A. Facchini, M. Ienca. *Finding consensus on trust in AI in health care : recommendations from a panel of international experts*. February, 2025, DOI: 10.2196/56306.

D. E. Carlson, **R. Chavarriaga**, Y. Liu, F. Lotte, B.-L. Lu. *The NERVE-ML (neural engineering reproducibility and validity essentials for machine learning) checklist : ensuring machine learning advances neural engineering*. March, 2025, DOI: 10.1088/1741-2552/adbfbf.

A.2 CAI Team as of 31.03.2025

Name	Function	Email
Dr. Ahmed Abdulkadir	Research Associate	abdk@zhaw.ch
Dr. Waqar Ali	Research Associate	aliw@zhaw.ch
*Erdal Ayfer	Intern, Bachelor Student	31.08.2024
Daniel Barco	Doctoral Student	baoc@zhaw.ch
Yann Billeter	Research Assistant, Master Student	bily@zhaw.ch
Prof. Dr. Jasmina Bogojeska	Senior Lecturer, Head of XAI Group	bogo@zhaw.ch
Dr. Hella Bolck	Research Associate, Head of Continuing Education	boce@zhaw.ch
Diego Bolliger	Doctoral Student	boie@zhaw.ch
*Oliver Carmignani	Research Assistant	31.12.2024
Dr. Ricardo Chavarriaga	Research Associate, Head of RAI Group, MSc Programme Coordinator	char@zhaw.ch
Prof. Dr. Mark Cieliebak	Senior Lecturer, Head of NLP Group	ciel@zhaw.ch
Maggie Deller Haight	Administrative Assistant	deel@zhaw.ch
Dr. Philipp Benedikt Denzel	Research Associate	denp@zhaw.ch
Dr. Jan Milan Deriu	Research Associate	deri@zhaw.ch
Dr. Manuel Dömer	Associated Faculty Member	doem@zhaw.ch
*Raphael Emberger	Research Assistant, Master Student	31.08.2024
Dr. Anna Fedorova	Doctoral Student	fedo@zhaw.ch
Oliver Forster	Research Assistant	foro@zhaw.ch
Prof. Dr. Rudolf Marcel Fuchsli	Associated Faculty Member	furu@zhaw.ch
*Xinrui Gao	Intern, NCCR Fellowship	31.07.2024
Dr. Elena Gavagnin	Associated Faculty Member	gava@zhaw.ch
Patrick Giedemann	Doctoral Student	gied@zhaw.ch
*Nicola Good	Research Assistant, Master Student	31.01.2024
Prof. Dr. Christoph Heitz	Associated Faculty Member	heit@zhaw.ch
*Marius Huber	Research Assistant	31.12.2024
Manuela Hürlimann	Research Associate	hueu@zhaw.ch
Dr. Deepak Ingole	Research Associate	inge@zhaw.ch
Benjamin Meyer	Doctoral Student	mebr@zhaw.ch
Katsiaryna Mlynchyk	Research Assistant, Master Student	mlyn@zhaw.ch
Martin Oswald	Research Assistant, Master Student	oswd@zhaw.ch
Melvin Pariyadan	Research Assistant, Master Student	pari@zhaw.ch
*Dr. Manuel Renold	Senior Lecturer, Head of CLS Group	31.12.2024
Dr. Alisa Rupenyan-Vasileva	Senior Lecturer, Head of IAI Group	rupn@zhaw.ch
Felix Saaro	Research Assistant, Master Student	saaf@zhaw.ch
Pascal Sager	Doctoral Student, Head of Demos	sage@zhaw.ch
Prof. Dr. Frank-Peter Schilling	Senior Lecturer, Deputy Director of Centre, Head of IVS Group, Head of PhD Programme in Data Science	scik@zhaw.ch
Lars Schmid	Research Assistant, Master Student	shmr@zhaw.ch
*Artina Sijarina	Intern, IAESTE	16.08.2024
Dr. Marc Stadelmann	Research Associate, Head of Infrastructure	stmd@zhaw.ch
Prof. Dr. Thilo Stadelmann	Senior Lecturer, Director of Centre, Head of MPC Group	stdm@zhaw.ch
*Wolfgang Stefani	Doctoral Student	30.04.2024
Samuel Stucki	Research Assistant	stku@zhaw.ch
Yu Tang	Doctoral Student	tanu@zhaw.ch
*Sonja Trinkner	Administrative Assistant	31.10.2024
Dr. Don Tuggener	Research Associate	tuge@zhaw.ch
Dr. Lukas Tuggener	Associated Researcher	Lukas.tuggener@rwai.ch

*Julia Usher	Doctoral Student	31.12.2024
*Janik Vollenweider	Research Associate	31.12.2024
Pius von Däniken	Research Assistant, Master Student	vode@zhaw.ch
Prof. Dr. Christoph von der Malsburg	Visiting Professor	malsburg@fias.uni-frankfurt.de
Dr. Joanna Weng	Associated Faculty Member	wenj@zhaw.ch
*Mingkun Wu	Intern, NCCR Fellowship	31.07.2024
Peng Yan	Doctoral Student	yanp@zhaw.ch

* = With CAI until listed date

A.3 Location

CAI Centre for Artificial Intelligence

Technikumstrasse 71
PO Box
CH-8401 Winterthur
www.zhaw.ch/en/engineering/institutes-centres/cai/

Contact

Prof. Dr. Thilo Stadelmann
Phone +41 58 934 72 08
thilo.stadelmann@zhaw.ch

Administration

Maggie Deller Haight
Phone +41 58 934 76 48
office.cai@zhaw.ch



Zurich University
of Applied Sciences

School of Engineering

**CAI Centre for
Artificial Intelligence**
Technikumstrasse 71
PO Box
CH-8401 Winterthur
www.zhaw.ch/cai/en

Contact

Prof. Dr. Thilo Stadelmann
Head of Centre
+41 58 934 72 08
thilo.stadelmann@zhaw.ch

Administration

+41 58 934 76 48
office.cai@zhaw.ch