Advancing the Sustainability of Machine Learning and Artificial Intelligence via Labeling and Meta-Learning

Dissertation

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

der Technischen Universität Dortmund
an der Fakultät für Informatik

von

Raphael Fischer

Dortmund

2025

Tag der mündlichen Prüfung:	01. Juli 2025
Dekan:	Prof. Dr. Jens Teubner
Gutachter*innen:	Prof. Dr. Thomas Liebig
	Prof. Dr. Geoffrey I. Webb

Abstract

Artificial Intelligence (AI), driven primarily by advances in machine learning, is having a transformative impact on our world. While the availability of AI offers numerous technological opportunities, it also poses significant challenges to our society, economy, and environment. Despite the imperative need for sustainable development, the research community and service providers are mostly focused on scale and predictive quality, while neglecting the importance of resource efficiency and transparency. This observation is particularly problematic in the context of AI-as-a-service, as modern AI practitioners can neither be expected to understand intricate performance trade-offs nor make sustainable decisions. This dissertation addresses the challenge of advancing AI sustainability by establishing more transparency and aiding informed decision making, under consideration of diverse stakeholder perspectives. The thesis first introduces fundamental concepts of AI and discusses the vast research landscape in which it is situated. It then presents three central contributions based on respective scientific publications: (1) a methodology and software framework for sustainable and trustworthy reporting (STREP), (2) concepts and evaluations for the high-level labeling of AI models, and (3) a novel take on meta-learning and automated machine learning that allows for being resource-aware and user-centric. After critically reviewing the shortcomings of current reporting, the STREP methods are introduced and applied to investigate performance trade-offs and reporting biases regarding AI models and hardware. Inspired by consumer communication systems such as energy labels, concepts for labeling AI models are proposed and validated through interdisciplinary thematic analysis. Finally, the thesis extends the general idea of meta-learning to perform automated model selection while accounting for multiple performance dimensions and user-defined priorities. For all three contributions, the theoretical formulations are empirically evaluated through experimental investigations, spanning learning domains such as computer vision and time series forecasting, as well as different hardware setups including powerful deep learning processors or low-power edge devices. The accompanying software repository provides additional benefits to readers and practitioners, offering generalized implementations of the central STREP methodology and an interactive exploration tool for all experiments. The thesis concludes with a critical discussion of its findings, limitations, and directions for future research on AI sustainability. By proposing means for bridging knowledge gaps and explicitly considering resource efficiency during model creation, this work promotes sustainable development in the evolving AI landscape.

Acknowledgments

This thesis would not have come into existence without the much-appreciated support and help of a wide range of people, that I want to thank from the bottom of my heart. Your unwavering faith in me at times surpassed my own faith in this work—you are an invaluable source of strength and perseverance! Of course, naming everyone is not feasible, but I would at least like to shortly introduce a representative sample.

First of all, thank you to Thomas Liebig, who provided me with excellent input for the completion of this work and was a fantastic supervisor despite all circumstances. Thank you also to Katharina Morik, who kindly introduced me to the world of research and contributed a lot of important ideas and support throughout the years, and to Geoff Webb, who not only guided the writing of this thesis but whose welcoming nature actually motivated me to pursue a PhD in the first place. Thank you also to Jakob Rehof and Christian Janiesch, who completed my committee and always gave helpful feedback on my academic work, and Jian-Jia Chen, who supported me as a mentor. The whole LS8 | ML2R | Lamarr team offered the most wholesome working environment—together, we went through a lot of change, and while I enjoyed the company of all my colleagues on this journey, I would like to especially thank Ann-Kathrin, Sascha, Matthias, and Sebastian for always having my back, during and after work hours, and through all the ups and downs! I want to also thank David and the rest of the Wilo team for the fruitful exchange during our collaboration. Finally, a big thank you goes out to all the remaining co-authors and (research) colleagues that I was fortunate enough to collaborate with—you provided me with fascinating new perspectives and I could not have done this without you!

Of course, my family and friends also made sure that I turned off the screen from time to time, helping me maintain a healthy level of social well-being during my PhD. As such, my biggest gratitude goes to Sabine, Michael, and Miriam for guiding every step and decision in my life, to Vanessa for all your love and support, to Yannick and Frederik for countless late night hours in shared hobby projects, and to Marius, Joline, Robin, Nils, Caro, Mino, Chris, Phil, Kevin, Vero, Viktoria, Babsi, Matthias, and all the other friends and music colleagues who accompany me on my path. Whether it be at social home hangouts, club nights, physical exercises, music production, festival weekends, or vacations, you always succeed in demonstrating how well my extrovert nature complements my rather rational work life. Thank you for being the gems you are and supporting me wherever you can!

There are a lot of other names on my mind, so this should not be considered to be an exhaustive list. To all the amazing people in my life: I can hardly express how happy I am to have you on my side!

Contents

Glossary

ARC *Abstraction and Reasoning Corpus*. 15, 16, 44, 45

PWC *Papers With Code*. 45, 48, 57, 58, 68, 69, 122

STREP Sustainable and Trustworthy REPorting. 8, 10, 11, 43, 44, 47, 48, 50, 51, 53–60, 62, 63, 65, 68, 70, 72–77, 81, 84, 85, 89, 93, 95, 112, 115–118, 121, 122, 125, 127

AB *Adaptive Boosting*. 23, 106

ACC1 average top-1 ACCuracy. 25, 27, 48, 53, 59, 60, 67

ACC5 average top-5 ACCuracy. 25–27, 53, 67

AGl `AutoGluon`. 100–102, 105, 106, 110, 112

AI Artificial Intelligence. 5–19, 27–29, 34–46, 48, 50–55, 61, 62, 69–73, 75–92, 98, 106, 113, 115–123, 127

AIaaS AI-as-a-Service. 5, 8, 18, 27–29, 38, 41, 43, 51, 71, 75, 79, 80, 83, 88, 89, 115, 119

AKe `AutoKeras`. 100–102, 105

AP Average Precision score. 25, 26

API Application Programming Interface. 44, 45, 48

AR Average Recall score. 25, 26

ASk `Auto-Sklearn`. 100–102, 105

AutoML Automated Machine Learning. 8, 9, 11, 12, 15, 17, 22, 30–33, 38, 42, 57, 91, 92, 99–101, 103–106, 110–113, 115–118, 120, 123–125

CML Compositional Meta-Learning. 9, 12, 70, 91, 92, 94–106, 108–113, 115–118, 123–125, 127

CPU Central Processing Unit. 7, 24, 28, 40, 41, 49, 63–66, 122

CV Computer Vision. 5, 17, 19, 21, 28, 38, 58, 62, 64, 70, 122

DL Deep Learning. 5, 12, 18, 23, 27, 28, 32, 33, 59, 62, 63, 75, 99, 121

DML Direct Meta-Learning. 99, 104

DNN *Deep Neural Network*. 23–25, 27–33, 35, 42, 57, 59, 62, 63, 65, 75, 77, 91, 92, 99–106, 110, 112, 113, 117, 122, 123

ENI ENergy draw during Inference (per batch). 25, 26, 41, 59–61, 63, 64, 67, 106, 122

ENT ENergy draw of Training. 25, 26, 41, 48, 60, 101

ERM Empirical Risk Minimization. 20–22

EU European Union. 6, 36, 44, 52, 61, 75, 77

F1 average F_1 score. 25, 26

FLOP FLoating-point OPeration. 24–26, 28

FS File Size of model on disk. 25, 26, 60, 67, 100

GenAI Generative Artificial Intelligence. 5, 18, 19, 27–29, 36, 38, 39, 41, 45, 46, 71, 75, 79, 83, 91, 115

GNB *Gaussian Naive Bayes*. 22, 106–108, 123

GPU Graphical Processing Unit. 7, 24, 25, 28, 32, 40, 41, 49, 60, 62, 79

HPO Hyperparameter Optimization. 30–32, 99, 113, 119

IID Independent, Identically Distributed. 21

IT Information Technology. 35, 38, 81

kNN *k-Nearest Neighbor*. 23, 106, 107, 123

LBP *Loopy Belief Propagation*. 22, 76

LogR *Logistic Regression*. 22, 106

LR *Linear Regression*. 22

MAE Mean Absolute Error. 25, 26, 95, 96

mAP50 intersection over union mean AP with 0.5 threshold. 25, 26

mAP95 mAP with various thresholds between 0.50 and 0.95. 25, 26

MAPE Mean Absolute Percentage Error. 25, 26, 69

MASE Mean Absolute Scaled Error. 25, 26, 69, 101–105, 123

ML Machine Learning. 5–12, 14–24, 26–35, 38–44, 46, 48, 49, 51, 55, 58, 65, 69, 71–76, 79, 80, 83, 86, 88–92, 94, 95, 97, 106–108, 110, 112, 113, 115–119, 121–123

MLP *Multilayer Perceptron.* 23, 27, 106

MP number of Model Parameters. 25, 26, 53, 60, 67, 100

MRF *Markov Random Field*. 16, 22, 76, 77, 122

NAM `Naive AutoML`. 106, 110

NAS Neural Architecture Search. 31, 32, 99, 110, 113, 119

NCS Neural Compute Stick 2. 63–65, 122

NFL No Free Lunch. 30, 50, 53, 69, 94, 117

NLP Natural Language Processing. 5, 19, 27, 28, 38, 42, 44

PFN `TabPFN`. 106, 110–112, 124

QR Quick Response. 62, 76, 77, 90

RAM Random-Access Memory. 98, 106

RF *Random Forest*. 23, 106, 108

RL Reinforcement Learning. 19, 32

RMSE Root Mean Squared Error. 25, 26, 105

RR *Ridge Regression.* 22, 106, 108

RTI Running Time during Inference (per batch). 25, 26, 48, 53, 60, 61, 63, 64, 67, 106, 122

RTT Running Time of Training. 25, 26

SD Sustainable Development. 5–8, 10, 12, 24, 27, 34, 36–39, 43, 85, 90, 97, 113, 117, 120

SDG Sustainable Development Goal. 5, 36, 37, 39

SGD *Linear Stochastic Gradient Descent*. 22, 106, 107

SVM *Support Vector Machine.* 22, 73, 106–108

TPU Tensor Processing Unit. 25, 63–65, 122

USB Universal Serial Bus. 7, 13, 41, 56, 62–65, 117, 122

XAI eXplainable Artificial Intelligence. 15, 34, 35, 46, 71, 87

XRF *eXtra Random Forest*. 23, 106

1 Introduction

Artificial Intelligence (AI), a vision that dates back to ancient stories and myths [MC04], has matured into being ubiquitous in academia and business. This evolution was primarily enabled by advances in Machine Learning (ML) research, a field that was only initiated 70 years ago [Sam59; McC+55] but has continuously grown in size and literature output, recently surpassing 100 000 yearly ML publications [Web25]. The vast methodological repertoire of ML, and Deep Learning (DL) in particular [LBH15], allows to train and deploy powerful AI models for domain applications such as Computer Vision (CV) [Sze11], Natural Language Processing (NLP) [JM25], or protein folding [JT22]. While not having reached human-level intelligence [Alt+24], the capabilities of modern Generative Artificial Intelligence (GenAI) and foundation models are impressive [Feu+24]: They break the Turing test [Bie23] and enable several application scenarios for human-AI interaction, for example in the context of industry 5.0 [MBC24]. Additionally, the AI hype is boosted by the growing accessibility thanks to AI-as-a-Service (AIaaS) [ES20], referring to the use of pre-trained models in the cloud instead of assembling local expertise and hardware for performing ML.

However, the opportunities afforded by modern AI do not come without costs—quite the contrary, the increasing availability in combination with the observable compute trends [Sev+22] are highly distressing in the context of the well-established need for Sustainable Development (SD), as centrally manifested by the United Nations *Agenda 2030* [Col15]. The associated Sustainable Development Goals (SDGs) aim at balancing the needs of the economy, environment, and social well-being, however global issues like ongoing conflicts and climate change [EA24] cast a gloomy light on the current state of the world, and in particular, the five years left for advancing the SDGs. Regarding ML and intelligent services, it is important to acknowledge two perspectives on sustainability [Wyn21]: On the one hand, AI can potentially benefit the pursuit of SDGs [KCS22], for example in the context of managing natural disasters [Int24b] or facilitating a circular economy [KCS22]. On the other hand, however, AI also poses a threat to all three dimensions of sustainability, for example evidenced by "unrivaled inequality and environmental degradation" [Sæt21] caused by surveillance capitalism [Zub18].

1.1 Motivation

Specific research areas have emerged for explicitly investigating the negative implications of AI for the economy, environment, and society, for example investigating the explain-

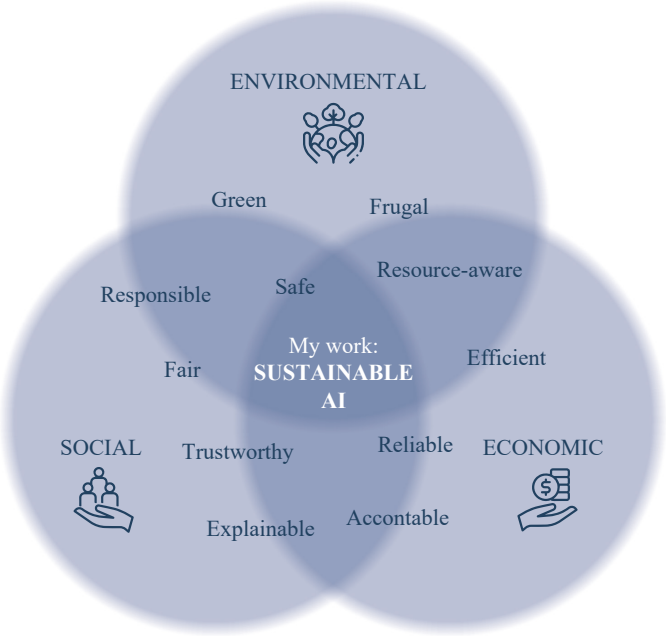

Figure 1.1: The three dimensions of sustainability in the context of AI, acting as an umbrella framework for the various adjacent research areas that discuss the role of AI in relation to environment, society, and economy.

ability [Sam+19; Lan+21], ethics [Flo+18], trustworthiness [Cha+21], safety [Ben+25], or responsibility [Dig19] of AI. For example, the recently published *International AI Safety Report*, distilled from the opinions of 100 experts in the field, highlights the imminent "systemic risks" of modern AI with respect to labor markets, market concentration, environment, and privacy [Ben+25]. This also naturally fueled the first wave of AI regulations, the best-known examples being the European Union (EU) AI Act [PE24] and the (by now repealed) United States presidential executive order [Bid23]. While it is hard to cleanly separate the different research areas from one another, they can easily be brought together under the central concept of AI sustainability, as visualized in Figure 1.1. Similar connections have been made in several other works [Cha24; Roh+24; GM22; Tru20], and since "Trustworthy AI" and "Resource-aware ML" are prime research areas at the *Lamarr Institute* [Lam25], I dedicated my PhD to advancing research on AI sustainability, linking both directions.

While the pressing matter of global climate change and the precarious role of AI for SD form the main motivation for my research and this thesis, the specific focus lies on the second perspective on AI sustainability, thus following the urgent call to make the field itself more sustainable [Wyn21]. In numerous works, experts have argued that large parts of the ML community are overly focused on improving quality metrics for specific learning tasks at high computational costs [LJS24; Bir+22; Sev+22]. Advances toward ML efficiency, for example in the form of model compression [Cho+20], ensemble pruning [TPV09], or

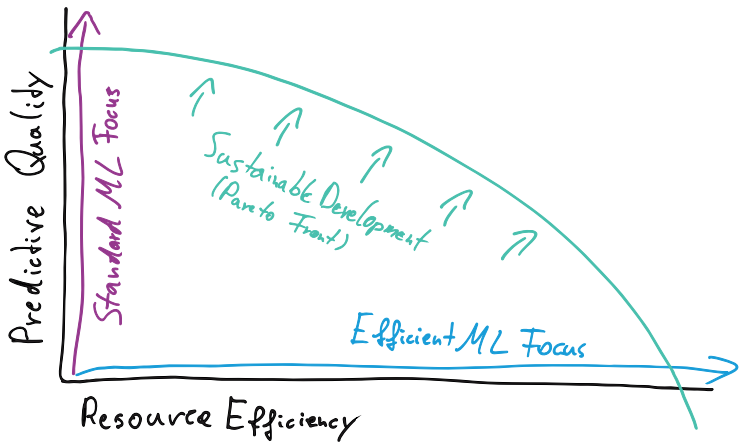

Figure 1.2: Visual illustration of the relation between predictive quality and resource efficiency in ML. Advancing the field requires to acknowledge and advance the resulting Pareto front [VLW25].

memory layout optimization [Bus+18], are less frequent. SD requires to explicitly balance predictive quality with resource consumption, acknowledging and advancing the resulting Pareto front, as shown in Figure 1.2, inspired by the argumentation of Varoquaux et al. [VLW25]. Nevertheless, developing and using AI in sustainable ways might sound straightforward at first, but is highly non-trivial: First of all, measuring AI sustainability is not easy in itself, as environmental costs stem from the complete life cycle [Wu+22], and thus, are subject to various factors like data, algorithms, software, hardware, and carbon intensity [Luc+19]. Indeed, later experiments will showcase how practical properties of AI models, even in relative comparisons, significantly change with deployment across different hardware setups, like Central Processing Units (CPUs), Graphical Processing Units (GPUs), or specialized AI hardware like Universal Serial Bus (USB) edge accelerators [Reu+19]. The diversity of AI users and stakeholders moreover necessitates to view this problem not only from a technology-savvy perspective, but also break down ML efficiency information for transparently addressing a broader target audience [Luc+25].

As an example to illustrate these problems, imagine how AI solutions might be practically developed in the application domain of industrial manufacturing [Sol24]. Naturally, addressing any business opportunity or use case requires to connect domain experts (e.g., mechanical engineers, product designers, machine operators) with the ML experts, which can develop AI solutions based on available data. Moreover, several other stakeholders will be involved, for example to align innovation with business strategies (management), to correctly follow law and regulations (legal department), and to make maximum profit with innovative products (marketing and sales). The practical behavior of ML models and the resulting implications for business use are however extremely intricate, which complicates the development of sustainable AI solutions. Note that this example is not only of hypothetical nature—it is actually based on my experience from working in a strategic research partnership with *Wilo SE*, a Dortmund-based water pump manufacturer.

Over the course of three years, we developed various AI prototypes for domain-specific use cases [Fis+23], always paying close attention to resource efficiency because sustainability forms "the overall framework for the corporate strategy" at *Wilo* [Wil24]. A significant amount of time was spent on bridging knowledge and communication gaps between stakeholders similar to those described above, which is consistent with the general realization that "there is only a small fraction of time for data analysts and scientists to do analysis work" [ADW17]. Considering the aforementioned trend toward AIaaS and the reduced necessity to include ML experts during development, such gaps become even more problematic [Bre+23].

Based on my personal experience and the related literature, I therefore argue that two specific issues need to be addressed in order to promote and drive SD in the context of ML and AI:

I1 - *Transparency:* ML methods and their implications are hard to understand, yet the resulting models are available to be used by nearly anyone. Today's practitioners cannot be expected to learn about all complex intricacies, but bridging the knowledge gaps is of central importance for developing and using AI in sustainable ways [Roh+24]. As a result, it is necessary to rethink and improve the current state of communicating scientific results and reporting on AI model behavior.

I2 - *Decision Making:* As there is "no free lunch" [WM97], practitioners need to make difficult decisions during AI development and deployment. Automated Machine Learning (AutoML) aims at easing these decision processes, however so far, does neither adequately consider sustainability, nor take users' individual needs into account [Tor+23]. Novel approaches are needed for automating the model selection during AI development, while paying close attention to SD dimensions.

Note that these two issues are closely linked: First, properly understanding the problem and possible solutions is an important prerequisite for informed decision making, and second, any decision in AI development, whether it be automated or manual, should also be made transparent and comprehensible.

1.2 Contributions

For advancing the sustainability of ML and AI, the thesis at hand addresses the identified issues via three central contributions, which are substantiated by peer-reviewed papers and summarized as follows:

C1 - *Sustainable and Trustworthy Reporting:* Current AI reporting hinders SD, therefore I introduce a methodological framework for Sustainable and Trustworthy REPorting (STREP) [FLM24]. It is also practically implemented as a software framework, which allows me to assess and compare the multi-objective performance and resource efficiency of AI models [Fis+22]. Moreover, the methods are applied to investigate

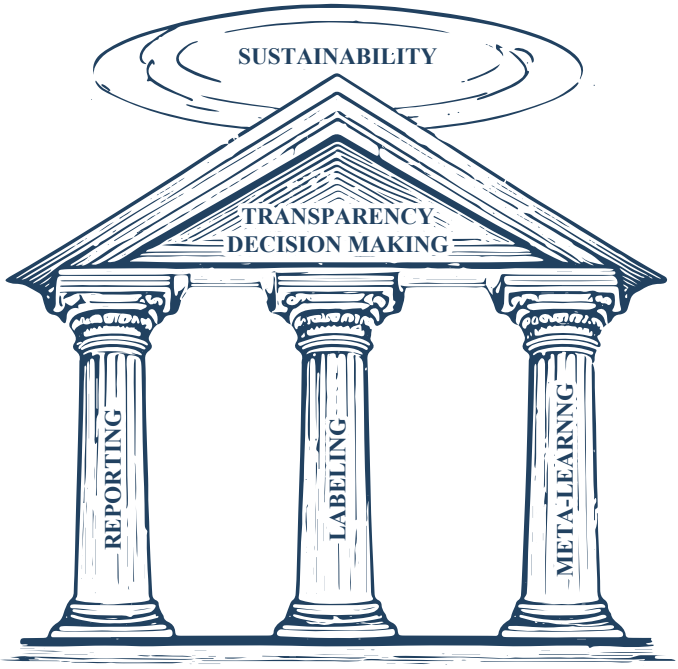

Figure 1.3: Illustration of the thesis motivation, visualizing how the three central contributions (pillars) support and benefit transparency and informed decision making in order to address the overarching goal of sustainable AI development and use.

the efficiency of specialized edge accelerator hardware [SFB24] and to unveil the biases in established reporting.

C2 - *AI Model Labeling:* In order to be more transparent, bridge knowledge gaps, and ease the communication about ML, I present and validate the concept of labeling AI models for practical use [Mor+22; Mor+21]. This novel reporting format allows to quickly learn about the intricate effects and trade-offs occurring in ML, thus benefitting a wide range of AI stakeholders and enabling them to make more informed decisions [Fis+25].

C3 - *Sustainable Model Selection via Meta-Learning:* To make the automation of model selection decisions during AI development more transparent and resource-aware, I present a novel extension to the meta-learning framework [Fis+24]. The proposed Compositional Meta-Learning (CML) enables users to infuse AutoML with their own preferences, understand the model recommendation on multiple levels, and thus align the development of AI with sustainability goals [FS24].

Figure 1.3 visually illustrates how the contributions (pillars) support the goal of establishing more transparency (I1) and aiding decision making (I2), in order to make AI development and use more sustainable. As with the addressed issues, the three contributions of my work are also interconnected along several lines: AI labels (C2) are later introduced as a key element for better reporting (C1), for communicating meta-learning results (C3)

is it important to also acknowledge different stakeholders' perspectives and levels of expertise (C2), and trustworthiness (C1) is later shown to be a central discussion point in the context of labeling (C2). The `STREP` methods (C1) are fundamental to C2 and C3, and the corresponding software framework is a central information source that allows for interactively investigating all experimental results at `https://github.com/raphischer/strep`.

It should be noted that my thesis presents holistic contributions that benefit the complete field of ML and AI research and development. As such, my work does not advance single ML methods or address very specific learning problems—it should rather be understood as a methodological toolkit that can be applied in any learning domain, establishing more sustainability by fostering transparency and aiding decision making. This is further evidenced by the various application domains and experimental setups discussed in this thesis, which demonstrate the practicability of my methods across images, time series, and tabular data. While also contributing several theoretical formalizations, I primarily envision my thesis to be a practical resource that helps practitioners developing or using AI in sustainable ways. As such, theoretical aspects and guarantees of ML are only tangentially mentioned, and more emphasis is placed on providing practice-related insights.

While the overall goal of advancing AI sustainability is naturally much bigger than what can be covered in a single doctorate thesis, the significance of my contributions is evidenced by several related works. To illustrate this statement with an example, the proposed "sustainability criteria and indicators for AI systems" [Roh+24] encompass documentation, transparency, and energy consumption, and my thesis makes direct contributions toward these matters. Building on the collaboration with various research disciplines and application domains, this work approaches AI sustainability from different interdisciplinary perspectives. The goal is hereby to enable practitioners to utilize and advance ML methods in trustworthy and resource-aware ways, and to help with solidifying the spirit of SD in the context of AI technology.

1.3 Thesis Structure

In the following, I shortly summarize each of the chapters and list my associated research papers, which were written and published during my doctorate and thus support my thesis as a whole. A more detailed description of my contributions toward the individual publications is given in Section 1.4.

Chapter 1 - Introduction My thesis starts with explaining the central motivation for my work, discussing the current issues in using and developing AI technology, and formulating my contributions for establishing more sustainability in the field. Moreover, a concise overview of my scientific record is given, both in terms of published literature that supports the main part of this thesis, as well as some additional works.

Chapter 2 - Background In the second chapter, all necessary fundamentals for understanding the concepts, methods, and investigations of my thesis are presented. First, it introduces general theory as well as practical considerations for understanding and using ML and AI methods and models. Based on these formulations, the second section then discusses central concepts and frameworks of model selection, AutoML, and meta-learning. The chapter finishes with a broader background on sustainability in the context of AI, including a discussion of adjacent research fields and practical thoughts on measuring sustainability and resource consumption.

Chapter 3 - Sustainable and Trustworthy Reporting Based on three publications, this chapter discusses methods for reporting ML results in a more sustainable and trustworthy way, and moreover demonstrates their practical feasibility. After analyzing the shortcomings of current reporting, the necessary methodology for better reporting is introduced, including concepts for *characterizing*, *index scaling*, *compound scoring*, *categorical rating*, and *analyzing correlations* of model performance results. All five concepts are implemented in the corresponding `STREP` software framework, which is also presented in the chapter and can be found at `https://github.com/raphischer/strep`. To foster reproducibility, the repository also entails scripts for generating all figures presented in this thesis. In practical investigations, I showcase the effectiveness of `STREP` by assessing and comparing model performance in a resource-aware way. Moreover, the methods are applied to discuss the resource efficiency of specialized edge accelerator hardware for AI. Last but not least, the introduced concepts of `STREP` are utilized to investigate a broad range of report databases in terms of their inherent sustainability and biases. Primarily covered publications:

Raphael Fischer, Thomas Liebig, and Katharina Morik. "Towards More Sustainable and Trustworthy Reporting in Machine Learning". In: *Data Mining and Knowledge Discovery* (2024). ISSN: 1573-756X. DOI: 10.1007/s10618-024-01020-3

Raphael Fischer et al. "A Unified Framework for Assessing Energy Efficiency of Machine Learning". In: *Machine Learning and Principles and Practice of Knowledge Discovery in Databases*. 2022, pp. 39–54. DOI: 10.1007/978-3-031-23618-1_3

Alexander van der Staay, Raphael Fischer, and Sebastian Buschjäger. "Stress-Testing USB Accelerators for Efficient Edge Inference". In: *Proceedings of the 9th Symposium on Edge Computing (SEC)*. 2024, pp. 1–14. DOI: 10.1109/SEC62691.2024.00015

Chapter 4 - AI Model Labeling As explained later, AI labeling is a key concept for `STREP` and specifically addresses non-experts, which require comprehensible and transparent means for communication. Due to the complexity of establishing AI labels, the topic receives its own dedicated chapter, in which I present the conceptual idea in all details and explain how the proposed labels have evolved over the years. To validate the theory of AI labeling for practicability, I also present results from an interdisciplinary qualitative user study. For this evaluation, a diverse group of practitioners was interviewed, allowing

me to analyze the limitations and benefits of AI labeling and as well as the relations to making AI more trustworthy. The chapter closes with an in-depth discussion of AI labels that also formulates four central guidelines for their future refinement. Primarily covered publications:

Katharina Morik et al. "Yes We Care! - Certification for Machine Learning Methods Through the Care Label Framework". In: *Frontiers in Artificial Intelligence* (2022). DOI: 10.3389/frai.2022.975029

Katharina Morik et al. *The Care Label Concept: A Certification Suite for Trustworthy and Resource-Aware Machine Learning*. 2021. URL: https://arxiv.org/abs/2106.00512

Raphael Fischer et al. "Bridging the Communication Gap: Evaluating AI Labeling Practices for Trustworthy AI Development". In: *Proceedings of the 2025 AAAI/ACM Conference on AI, Ethics, and Society*. (forthcoming). 2025. URL: https://arxiv.org/abs/2501.11909

Chapter 5 - Sustainable Model Selection via Meta-Learning Moving away from mere AI use toward AI development, the fifth chapter explores how the concept of meta-learning can be extended in sustainable ways. For that, I first introduce a formal methodology for understanding model selection as a multi-objective optimization problem. It is approached by the novel CML approach, which allows to (a) make meta-learning considerate of performance trade-offs, (b) align model selection with user preferences, and (c) accompany model selections with multi-level explanations. In the second and third sections of the chapter, the theory of CML is practically applied to two learning domains, namely time series forecasting with DL and tabular data classification with classical ML algorithms. For that, large-scale experimental evaluations were conducted to assemble the meta-learning databases *XPCR* and *MetaQuRe*, which contain resource-aware performance data for various model configurations. The practical investigations evidence that CML is not only competitive with state-of-the-art AutoML with respect to predictive capabilities, but is also much more resource efficient and interactive. As such, it is a powerful tool for the SD of AI, as it eases decision making and establishes more transparency. Primarily covered publications:

Raphael Fischer and Amal Saadallah. "AutoXPCR: Automated Multi-Objective Model Selection for Time Series Forecasting". In: *Proceedings of the 30th International Conference on Knowledge Discovery and Data Mining (KDD)*. 2024, pp. 806–815. ISBN: 979-8-4007-0490-1. DOI: 10.1145/3637528.3672057

Raphael Fischer et al. "MetaQuRe: Meta-learning from Model Quality and Resource Consumption". In: *Machine Learning and Knowledge Discovery in Databases*. 2024, pp. 209–226. ISBN: 978-3-031-70368-3. DOI: 10.1007/978-3-031-70368-3_13

Chapter 6 - Discussion The thesis concludes with an in-depth discussion, where the discussed contents are summarized in context of the overall scope and goal of this thesis, namely advancing the sustainability of AI. I also critically discuss the limitations of this

work and highlight several opportunities for future work, representing guidelines for further advancing the sustainable, trustworthy, and resource-aware development and use of AI systems.

1.4 Declarations

For the literature that substantiates the main part of this thesis, I want to give some extra information on my contributions as an author. In most works I was the first author, meaning that I conceptualized the research idea, performed the discussed experiments, and was primarily responsible for writing the manuscript. In most of these cases, the other authors contributed by adding specific text passages and providing me with valuable feedback on the concept, experiments, and manuscript.

The first outlying case is the joint work on "stress-testing USB accelerators" [SFB24]. While being second author in the peer-reviewed publication, I guided this research project from the beginning, wrote large parts of the original manuscript, and therefore was the first author in the respective preprint [FSB24]. As I had already shifted my focus to other projects, Alex offered to take over the submission and publication at the *Symposium on Edge Computing* 2024, for which I gladly offered the first authorship in return.

The initial "care label" works were a joint group effort that started in early 2021. The first paper [Mor+22] took some time until publication and features my name as the third author, directly after Katharina (who had the original idea and initiated the project) and Helena (who guided the early research on AI labeling). The extension to this first work is similar in its content and thus never left the preprint state [Mor+21], with me on the fifth author position. In both papers, nearly all of the authors were deeply involved in the process of writing and publishing, and I continuously advanced the labeling concept with several follow-up works [Fis+22; FLM24]. The most recent labeling evaluation study [Fis+25] was only finalized shortly before writing this thesis and was accepted for publication at the *Conference on AI, Ethics, and Society* 2025. In this work, Magdalena and me conceptualized the study and wrote the largest parts of the manuscript, Alex and Katha helped with the interview analyses, and Christian and Thomas contributed valuable feedback across all stages. By now, we have already published a follow-up study that explores reflective design theorizing in the context of AI labels [Sta+25].

I want to highlight that the chapters discussing C1–C3 were distilled from the respective research works and thus were unified in terms of terminology, formalizations, and experiments, resulting in minor differences to the original work. Any "verbatim quotes" will be denoted accordingly, however respective literature references will often only be given at the start or end of respective sections. Most parts of the thesis are written in passive or first-person plural perspective, either using the *inclusive "we"* to address potential readers of this work (e.g., "we now explore" in Chapter 3), or for highlighting results from collaborative work (e.g., "aligning with our research questions" in Chapter 4). The singular form will only be occasionally used, emphasizing my own opinions or personal takes and

ideas. Companies and brands like *Intel* will be emphasized with italics, whereas specific products and models such as `MobileNet` will be displayed in typewriter font [How+17]. I want to once again thank all co-authors for the effort that they have put into our joint publications—I can't express how much your expertise, feedback, and cooperation helped me to successfully complete my doctorate.

1.5 Additional Publications

During my time as a PhD student at *TU Dortmund University* and the *Lamarr Institute*, I was fortunate to explore a broad research landscape that exceeds the limits of a single thesis. While not discussing my other papers in detail, I still want to roughly outline these works (in descending order of publication date) and explain connections to the main part of my thesis.

"Prioritization of Identified Data Science Use Cases in Industrial Manufacturing via C-EDIF Scoring" (2023)

In 2022, a strategical research transfer partnership between the *Lamarr Institute* and the Dortmund-based water pump manufacturer *Wilo SE* was established. The prime goal was to identify and prototypically implement solutions for AI and ML use cases and business opportunities. The main findings from our use case exploration phase were summarized in the respective paper [Fis+23], in which we specifically presented a method to prioritize AI use cases based on their (E)valuability, (D)ata situation, (I)mpact, and (F)easibility, combined with the assignment (C)onfidence.

Although the work primarily discusses AI for industrial manufacturing, there are certain meaningful connections to the broader theme of this thesis. First of all, it evidences how AI can potentially benefit very specific application and business domains. As already discussed in Section 1.1, the work for this project demonstrated that identifying and tackling such use cases requires to bridge the communication and knowledge gaps toward the domain experts at *Wilo*. We did so via countless conversations and meetings, however novel communication formats like the later explored AI labels (Chapter 4) could potentially have alleviated some of these efforts. In addition, some applications demanded fast response times and resource-hungry solutions also directly cause higher costs, for example in terms of cloud compute power. As a result, we often found ourselves explicitly considering resource consumption and efficiency when implementing AI prototypes for *Wilo* use cases, which aligns with the central topic of resource-awareness in this thesis. To that end, the collaboration with *Wilo* actually turned out to be a perfect match, as "sustainability and social responsibility play an important role in all decision-making and business processes at Wilo" [Wil24].

"Energy Efficiency Considerations for Popular AI Benchmarks" (2023)

This short paper was an adaption of the `ImageNet` efficiency analysis [Fis+22] (to be discussed in Section 3.4.1) and used similar methods to explore the efficiency of multiple ML algorithms when applied to various tabular benchmark datasets [FJM23]. With over 100 experiment configurations, this work demonstrated that different data sets all have respective efficiency landscapes and also unveiled that some ML algorithms are more (or less) likely to act efficiently.

My work was positively received, accepted, and presented at the *AI for Energy Innovation* workshop at the Association for the Advancement of Artificial Intelligence (AAAI) conference in 2023. Unfortunately, the workshop organizers (in controversy to their initial call for papers) never issued any proceedings and moreover also never responded to my later mails, hence the paper remains in preprint state up to this date. Naturally, it strongly aligns with the goals of this thesis by investigating the environmental sustainability of AI methods and models. The experiments of this work later also inspired the experimental setup for assembling the `MetaQuRe` database [Fis+24], which will be explored in Section 5.3.

"Harnessing Prior Knowledge for Explainable Machine Learning" (2023)

This work was among the first collaborations across different partner locations within the *Lamarr Institute*. As a big team, we built upon the taxonomy of informed ML [Rue+23] and investigated how such prior knowledge is utilized in the context of eXplainable Artificial Intelligence (XAI) [Bec+23]. Our review categorized the literature into three primary approaches: embedding knowledge into the learning pipeline, enhancing explainability methods with prior knowledge, and deriving new knowledge from explanations to iteratively refine models.

While advancing XAI is not the central focus of this thesis, it is closely connected to the overall framework of sustainability—this relation will be discussed in more detail in Section 2.3.1. In the context of my contributions to the cause, labeling (Chapter 4) not only has the goal of explaining model properties, but model explainability is also explicitly considered in my work on sustainable meta-learning and AutoML (Chapter 5). Many use cases of the aforementioned *Wilo* partnership represent classic examples for informed ML that incorporates domain knowledge. This paper provided a wonderful opportunity to collaborate with fantastic colleagues and delve into XAI and informed ML literature.

"Solving Abstract Reasoning Tasks with Grammatical Evolution" (2020)

In the early stages of my doctorate, *Lamarr* colleagues and me participated in the *Abstraction and Reasoning Corpus* (*ARC*) competition [Fis+20b], which tests ML and AI capabilities for reasoning tasks with only minimal observed training data [Cho19]. While easy to solve for humans, these tasks required skills of abstraction that were non-trivial to implement

in ML models. We approached the problem by designing a domain-specific language for image transformations and applying grammatical evolution to search for possible task solvers. Our method demonstrated competitive performance, placing us among the top 4% of participants [Fis+20b].

While not really fitting the scope of this thesis, the *ARC* challenge is a perfect example for the limitations of modern AI. Unveiling how ML struggles with abstraction and reasoning naturally fuels AI skepticism and criticism, which is a central point of discussion in research fields like trustworthy and responsible AI, and thus, sustainability (the connections will be explained in Section 2.3.1). Moreover, the vast search space of task solvers was a central problem during the challenge—both for us, as well as for our competitors—and required to construct very efficient (i.e., resource-aware) solutions. Lastly, *ARC* represents a fine example of a benchmark that summarizes and compares the performance of different models. Among other types, reporting in the form of benchmarks will be critically discussed in Section 3.1.

"Probabilistic Gap Filling in Satellite Image Series" (2020)

Based on my master's thesis, this rather application-oriented paper discusses how clouds in images can be removed (i.e., how gaps can be filled) with the help of specialized spatio-temporal *Markov Random Field*s (MRFs) [Fis+20a]. They are a special representative of ML models, namely probabilistic graphical models, for which in-depth information can be found in Nico's excellent doctorate thesis [Pia19]. MRFs will be revisited in Section 4.1 because they served as the practical example for testing the original care label concept.

In the context of sustainability, this work can be seen as a useful contribution for improving remote sensing, which is important for tracking large-scale environmental developments. Even though I left the domains of remote sensing and probabilistic modeling in the early stages of my doctorate, this work acted as an important milestone. Visiting *Monash University* and collaborating with the wonderful team of Geoff Webb was a central motivation for pursuing a PhD in the first place. This paper manifested the contact and was my first experienced success as an AI scientist, helping me to withstand the struggles of scientific writing and publishing.

"Parameter Sharing for Spatio-Temporal Process Models" (2019)

This first publication of my doctorate was based on an early idea for my master's thesis, before it took a more application-oriented drift toward remote sensing imagery. The initial plan was to test whether methods of time series clustering could be feasible for compressing the aforementioned MRFs along their spatio-temporal dimensions [FPM19]. There is a clear connection to resource-awareness, and as such, sustainability, however I did not continue to work with probabilistic modeling and thus never advanced the idea any further.

2 Background

Exploring the central contributions of this thesis requires important fundamentals of ML and AI (Section 2.1), AutoML and meta-learning (Section 2.2), and sustainability in the context of AI (Section 2.3), which will be established in the following. Note that this background was deliberately limited to only introducing the concepts and discussion points that are relevant for understanding the later chapters, while the related literature enables interested readers to dive even deeper.

2.1 Machine Learning and Artificial Intelligence

Computer scientists perform *problem solving*, meaning that they traditionally formulate a problem and goal, search for a solution in the form of an *algorithm* (i.e., a sequence of mathematically computable instructions), and practically execute it via computers that consist of *hardware* and *software* [RN21]. As an example, detecting objects in images traditionally requires to combine various CV algorithms, which for example transform pixels, manipulate colors, or find edges via neighborhood filtering (i.e., convolutions) [Sze11]. Humans intuitively learn to visually identify objects and also solve many other complex problems, thanks to our strong abilities of comprehending and perceiving, representing characteristics of *intelligence* (derived from the Latin verb *intelligere*). As such, *artificial* intelligence (AI) has the goal of synthesizing human thought processes, reasoning, and behaviors [RN21], or as formulated during the first respective research workshop, "making a machine behave in ways that would be called intelligent if a human were so behaving" [McC+55]. Another early and highly influential work on AI suggests that AI would need to be indistinguishable from human intelligence with regard to having individual conversations via a computer, leading into what today is known as the *Turing test* [Tur50].

While traditional problem solving can provide somewhat intelligent solutions, it is often hindered by the complexity of problems—for example, in the context of detecting objects via CV, algorithmic adjustments are necessary to account for different types of objects, perspectives, and lighting. Instead of manually adjusting the solution toward the problem goal, computer scientists have therefore developed methods that solve the problem by instructing *machines* to *learn* from data about the problem, or in other words, ML [RN21]. Under the hood, performing ML corresponds to automatically tuning specific learning algorithms to the given data, thus combining statistical approaches with optimization techniques [HTF09]. The following will formalize learning problems and ML models

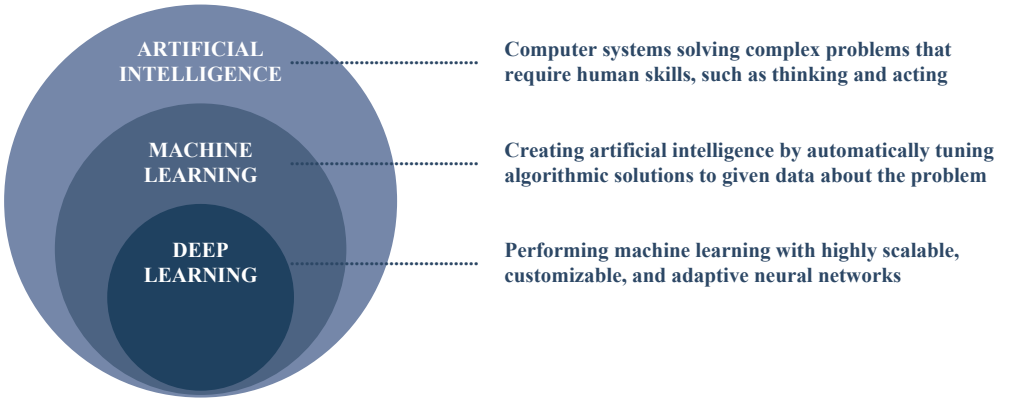

Figure 2.1: Connection of AI, ML, and DL, representing the most fundamental terms of this thesis.

(Section 2.1.1), investigate how they can be aligned with given data from the problem (Section 2.1.2), introduce various ML approaches (Section 2.1.3), and provide some additional information for putting ML to practice (Section 2.1.4). The specific subfield of DL constitutes specialized ML methods for learning particularly scalable and adaptive models, fueling an evolution that ultimately resulted in modern GenAI capabilities and AIaaS, to be further explored in Section 2.1.5. The connections between AI, ML, and DL are visually summarized in Figure 2.1, representing my own take on this widely known diagram summary. Note that in the course of this thesis, the ML term is used when referring to specific methods and algorithms, while AI either describes specifically learned models or non-specific, high-level phenomena concerning intelligent systems and services. While large parts of this section are inspired by related literature [RN21; HTF09], note further that the focus, terminology, and definitions were aligned with the later chapters of this thesis.

2.1.1 Learning Tasks and Models

This thesis mostly focuses on using ML for *supervised learning*, which comprises a specific set of *learning tasks* or problems that can be defined as follows:

Definition 2.1. *Let $\mathcal{X}_T$ and $\mathcal{Y}_T$ be the respective input (feature) and output (label) spaces of a supervised learning task T. Let further $f_T : \mathcal{X}_T \to \mathcal{Y}_T$ be an unknown function generating outputs $y \in \mathcal{Y}_T$ from given multidimensional inputs $\boldsymbol{x} \in \mathcal{X}_T$, i.e., $y = f_T(\boldsymbol{x})$. Supervised ML methods aim at building computational models $m \in \mathcal{M}_T$, which map features onto predicted labels $\hat{y}$ while closely approximating the true function f_T, i.e., $m \overset{!}{\approx} f_T$.*

To illustrate this with an example, consider a supervised binary image classification task (T), in which a computer shall categorize images $\boldsymbol{x} \in \mathbb{R}^n$ (i.e., n-dimensional vectors) based on whether they display dogs ($y = 0$) or cats ($y = 1$). As a possible instantiation of a function f_T, humans could inspect images $\boldsymbol{x}$ and assign respective labels y. For

single images, CV algorithms could be hand-tuned to successfully detect the specific animal [Sze11]. However, such algorithmic adjustments for correctly classifying large amounts of images are much easier to perform via numerical optimization, or in other words, ML. It requires methods for finding an algorithmic model $m \in \mathcal{M}_T$ that predicts labels $\hat{y}$ for given images $\boldsymbol{x}$ while closely mimicking the ground-truth function f_T, thus building an AI for the reasoning capabilities that humans use when inspecting images.

With this example in mind, note that the task T induces the type and structure of data within the input and output spaces. As such, single n-dimensional feature vectors $\boldsymbol{x} \in \mathcal{X}_T$, with $\mathcal{X}_T \subseteq \mathbb{R}^n$ could for example represent tabular information (no specific structure), time series (temporally ordered sequences of data), or images (two-dimensional arrays of possibly multi-dimensional pixel values). Likewise, label instances $y \in \mathcal{Y}_T$ could be numerical (commonly referred to as regression tasks, with $\mathcal{Y}_T \subseteq \mathbb{R}$) or categorical (classification tasks, with $\mathcal{Y}_T \subset \mathbb{N}$ entailing a set of ordinal encodings for specific categories). In many cases, the inputs $\boldsymbol{x}$ are only associated with one-dimensional labels y, however multi-label classification or higher dimensional regression tasks exist. As an example, time series forecasting requires temporally ordered data and has the goal of mapping an observed context $\boldsymbol{x}$ onto the possibly multidimensional forecast horizon $\boldsymbol{y}$ [Als+22]. Univariate time series entail a single evolving value, while multivariate series feature a vector of data at each point in time. Time series are commonly used to describe real-world evolving phenomena like human life or weather [God+21], and moreover, have strong connections to NLP, where transformation techniques such as *tokenization* and *word embeddings* allow to represent captured language and texts as multivariate time series [JM25]. Note also that many ML classification methods are not actually designed to only predict a single $\hat{y}$, and instead output estimated probabilities $\hat{\boldsymbol{y}} = (\hat{y}_i)_{i=1}^C$ for each of the C classes. These outputs can then be easily converted into a single class prediction via selecting $\hat{y} = \arg\max_{i=1}^C \hat{y}_i$. In some cases, the nature of the task and data at hand also restricts the applicable methods and models [Fis+22]—as examples, classification methods are not designed for solving regression tasks, and methods for time series forecasting should not be used for modeling image data. That said, many extensions have been proposed to generalize specific ML methods and facilitate their application to diverse learning tasks and data domains.

It is important to remember the specific terminology of Definition 2.1: Whenever this thesis mentions ML *methods*, it refers to specific concepts or algorithms for deriving and using ML *models*, which themselves can be understood as functional entities that map $\mathcal{X}_T$ onto $\mathcal{Y}_T$ while approximating the unknown function f_T. While this thesis mostly discusses ML in the context of supervised learning, it should also be mentioned that several other learning tasks exist. In unsupervised learning, the data comes without any observed labels y, and ML instead has the goal of identifying unknown patterns within $\mathcal{X}_T$ (which possibly could be labeled in a next step) [HTF09]. The mixture of both cases is known as semi-supervised learning, where only some instances $\boldsymbol{x}$ are annotated with labels. In self-supervised learning, a so-called pretext task is defined that allows to learn from vast amounts of unlabeled data in a supervised manner [Bal+23]—this concept fueled the evolution of GenAI services, which will be discussed in Section 2.1.5. The special case of Reinforcement Learning (RL) does not require any pre-assembled data or labels

and instead learns a policy by testing actions in a controlled environment, resulting in experience (data samples) to learn from.

2.1.2 Model Training

For now, we simply acknowledge that ML models can make predictions for observed inputs, or as given in Definition 2.1, obtain $\hat{y} = m(\boldsymbol{x})$. While Section 2.1.3 will later introduce different approaches to predict $\hat{y}$, we first explore how models can be aligned with observed data generated by the unknown function, or formally, how the goal of $m \overset{!}{\approx} f_T$ can be met. To that end, all ML models are *parameterized* in some way:

Definition 2.2. *Let $\boldsymbol{\theta} \in \mathbb{R}^{|\boldsymbol{\theta}|}$ be the real-valued parameters of a model $m_{\boldsymbol{\theta}} \in \mathcal{M}_T$. For any given input $\boldsymbol{x} \in \mathcal{X}_T$, they control how the input is processed and mapped onto the output prediction $\hat{y} = m_{\boldsymbol{\theta}}(\boldsymbol{x})$, with $\hat{y} \in \mathcal{Y}_T$.*

This definition adds another important aspect to the terminology of this thesis—whenever discussing models m, they are implicitly characterized by some specific parameter assignment $\boldsymbol{\theta}$, however the explicit notation $m_{\boldsymbol{\theta}}$ might be discarded in favor of comprehensibility. Parametrization represents the key difference to traditional problem solving, as it offers a means to numerically tune the algorithm behavior to the available data from the task, commonly referred to as *model training*. In supervised learning, it requires an observed *training dataset* and a *loss function*:

Definition 2.3. *Let $D_T = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$ be a training dataset with N learning examples, i.e., $D_T \subset \mathcal{X}_T \times \mathcal{Y}_T$. Let further $m_{\boldsymbol{\theta}} : \mathcal{X}_T \to \mathcal{Y}_T$ be a parametrized model to train on D_T, and $l_m : \mathbb{R}^{|\boldsymbol{\theta}|} \times \mathcal{Y}_T \times \mathcal{Y}_T$ be the model loss function describing the estimation error between ground-truth labels y and model predictions $\hat{y}$, given a specific set of parameters. Training the model $m_{\boldsymbol{\theta}}$ then corresponds to Empirical Risk Minimization (ERM) on D_T, i.e., finding good parameters $\boldsymbol{\theta}^*$ by solving the following optimization problem:*

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \frac{1}{N} \sum_{i=1}^{N} l_m(\boldsymbol{\theta}, m_{\boldsymbol{\theta}}(\boldsymbol{x}_i), y_i) \tag{2.1}$$

.

Going back to the earlier example of separating images into two classes $y \in \{0, 1\}$, one could calculate the loss for a model's predicted probability $\hat{y} \in (0, 1)$ as the logistic loss, i.e., $l_m(\boldsymbol{\theta}, \hat{y}, y) = -[y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})]$. By now, image classification tasks and datasets have however become much more complex, as can for example be seen from `ImageNet` [Den+09]. This "large-scale ontology of images" comprises hierarchical information on millions of images from the internet that come with corresponding labels for $C = 1000$ classes, including animals (e.g., magpie, scorpion, stingray), vehicles (airliner, ambulance, forklift), and various other objects (harp, kimono, sleeping bag). Compared to the binary case, such multi-class classification problems commonly use the cross-entropy

loss for training [HTF09], defined as $l_m(\boldsymbol{\theta}, \hat{\boldsymbol{y}}, y) = -\sum_{j=1}^{C} y \log(\hat{y}_j)$. Given the true label y, it aggregates the loss from the predicted probabilities $\hat{\boldsymbol{y}} = (\hat{y}_j)$ for each of the C classes. As a popular CV resource, `ImageNet` has allowed to train various image classification models such as `AlexNet` [KSH17], `VGG` [SZ15], `ResNet` [He+16a], or `MobileNet` [How+17], as well as several others [He+16b; Hua+17; Cho17; Gho17; San+18; Zop+18; How+19; TL19].

Looking at Definition 2.3 from a statistical point of view, computing the loss over a complete dataset can be understood as the empirical estimation of theoretical quantities [Yeh07]. Data instances $(\boldsymbol{x}, y)$ are actually originating from an unknown joint probability distribution $p_T(\boldsymbol{x}, y)$, for which the model $m_{\boldsymbol{\theta}}$ has an expected loss:

$$\mathbb{E}[L_m(\boldsymbol{\theta})] = \int l_m(\boldsymbol{\theta}, m_{\boldsymbol{\theta}}(\boldsymbol{x}), y)\, p_T(\boldsymbol{x}, y)\, d\boldsymbol{x}\, dy \tag{2.2}$$

The central assumption behind ERM is that the observed training data D_T is a random but representative sample from this distribution, meaning that it contains N Independent, Identically Distributed (IID) samples. Following the law of large numbers, the respective empirical loss would thus converge to the expected loss, forming the basis of statistical consistency in ML [Yeh07]:

$$\lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} l_m(\boldsymbol{\theta}, m_{\boldsymbol{\theta}}(\boldsymbol{x}_i), y_i) = \lim_{N\to\infty} \hat{L}_m(\boldsymbol{\theta}, D_T) = \mathbb{E}[L_m(\boldsymbol{\theta}))] \tag{2.3}$$

It is however important to note that the IID assumption is not guaranteed to hold in practice—putting too much confidence in the sample representativity can quickly result in *overfitting* the model on the training data, and observing unexpected behavior for inputs that were not seen during training. It is therefore common practice to hold back some of the available data during training and later use this *validation data* to assess the trained model's performance on unseen data [HTF09]. Accurately modeling the training data at hand while also maintaining a certain level of generalization on unseen data represents a balancing problem, which is commonly referred to as the *bias-variance trade-off* in ML that is also naturally connected to the data complexity and model expressivity [HTF09].

Looking back on Equation (2.1), we still have to discuss how the optimization problem of finding good parameters can be approached. Various optimizer options are available to fit the model parameters to the data by exploring and navigating the landscape of the loss function. Gradient-based approaches are commonly used to find good parameters via the derivative of Equation (2.1), such as the classic gradient descent and stochastic extensions for larger datasets [BV04]. Other approaches have used Bayesian optimization and evolutionary algorithms for training ML models without the use of gradients [BS93]. One should also keep in mind that training ML models often results in local optimality, despite techniques like momentum algorithms, multi-start methods, and stochastic optimization [Sut+13]. Non-trainable parameters, the so-called hyperparameters [FH19], further

control the internal model logic and optimization—the effort of tuning them represents a classic motivation for AutoML, which will be discussed in Section 2.2.1.

2.1.3 Methods and Algorithms

We still need to understand how parametrized models can actually make predictions, or as originally formalized in Definition 2.1, how feature vectors can algorithmically be mapped onto labels $\hat{y} = m_{\boldsymbol{\theta}}(\boldsymbol{x})$. Various respective ML methods have been developed, and while the algorithmic details are not relevant for understanding this thesis, the following introduces the most important concepts. Importantly, the upcoming explanations also indicate how the respective models are parametrized, however the generalized formalization of model parameters $\boldsymbol{\theta}$ will not be used, instead discussing weights $\boldsymbol{w}$ or other variables for calculating $\hat{y}$.

Solving problems via *linear models* was already common "in the precomputer age of statistics" [HTF09], assuming a linear dependency between features and labels. As such, a classic *Linear Regression* (LR) makes predictions via $\hat{y} = \boldsymbol{w}^\intercal \boldsymbol{x} + b$, based on a vector of feature weights (or coefficients) $\boldsymbol{w}$ and a bias (or intercept) b. Extensions to the LR approach exist for example in the form of *Ridge Regression* (RR), which imposes a penalty on larger coefficients via a so-called *regularization*, or *Logistic Regression* (LogR), which instead uses a logit function to predict probabilities for binary classification [Ped+11]. *Support Vector Machine*s (SVMs) also build on the idea of linear separation, however use a kernel function to also find non-linear decision boundaries via "constructing a linear boundary in a large, transformed version of the feature space" [HTF09]. While the basic versions of LogR and SVMs only allow for binary classification, they can also be extended for training multi-class models, for example via a *one-versus-rest* approach that breaks up the problem into multiple binary classification tasks [Ped+11]. As calculating gradients for training linear models on large datasets quickly becomes computationally expensive, additional variants such as *Linear Stochastic Gradient Descent* (SGD) have been developed [Ped+11].

Probabilistic ML methods approach the learning problem differently, by building empirical models for the underlying (unknown) probability densities for observing features and labels, thus linking back to the previously mentioned idea of ERM. *Gaussian Naive Bayes* (GNB) for example makes the pragmatic assumption that the n features are conditionally independent given the class label, and thus predicts the class probabilities as $\hat{y}_i = p(y) \prod_{j=1}^{n} p(x_j|y)$ [RN21]. MRFs, which were already mentioned in Section 1.5, follow a more sophisticated approach by modeling the joint probability distribution of all features via a graphical structure. As such, the probability of observing a specific instance $\boldsymbol{x}$ is factorized over the graph's clique potentials, leading to $p(\boldsymbol{x}) = \frac{1}{Z} \prod_{c \in \mathcal{C}} \psi_c(\boldsymbol{x}_c)$, with $\mathcal{C}$ containing all cliques, $\psi_c(\boldsymbol{x}_c)$ denoting the potential of a clique, and Z being a normalization constant known as the partition function. Different algorithms have been developed for practically performing probabilistic inference with MRFs, such as the approximative *Loopy Belief Propagation* (LBP) [Pia19].

As a very different and simple, yet effective approach, *decision trees* address learning tasks via a "greedy divide-and-conquer strategy" [RN21]. As such, they utilize a hierarchy of decision rules that describe the recursive subdivision of the input space $\mathcal{X}_T$ into M regions, with model parameters representing the binary decision logic in the tree nodes. Based on the input space division, predictions $\hat{y} = \sum_{m=1}^{M} \hat{y}_m \mathbf{1}(\boldsymbol{x} \in R_m)$ can be calculated as the average of the constant predictions $\hat{y}_m$ in each valid subregion, with $\mathbf{1}(\cdot)$ representing the indicator function for describing that $\boldsymbol{x}$ lies within region R_m (meaning that all rules for this region hold). As a non-parametric classifier, the *k-Nearest Neighbor* (kNN) approach also explores the input space, but aggregates the prediction from the labels of nearby data instances [RN21]. As such, it makes predictions via $\hat{y} = \frac{1}{k} \sum_{i \in \mathcal{N}_k(\boldsymbol{x})} y_i$, with $\mathcal{N}_k$ denoting the neighborhood for feature vectors $\boldsymbol{x}$.

The most prominent ML models to date arguably are *Deep Neural Network*s (DNNs), which have evolved from the conceptual idea of *Multilayer Perceptron*s (MLPs) [Ros58]. A single perceptron (i.e., neuron) simply uses a non-linear activation function ϕ to process the weighted inputs, or in other words, it calculates $\hat{y} = \phi(\boldsymbol{w}^\intercal \boldsymbol{x} + b)$. However, the model complexity is increased by arranging perceptrons into multi-layered networks [RN21], inspired by biological neural networks in neuroscience [MP43]. As such, MLPs make predictions $\hat{\boldsymbol{y}}$ by recursively calculating the outputs of the individual layers $l \in \{1, 2, \ldots, L\}$ via $\boldsymbol{h}^{(l)} = \phi^{(l)}(\boldsymbol{W}^{(l)} \boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)})$, with the trainable parameters corresponding to the weights and activations in the individual neurons. Over the years, the basic idea of MLPs was scaled up and expanded, eventually ushering in the current DL era that will be further discussed in Section 2.1.5 [LBH15]. All of the aforementioned `ImageNet` models belong to the family of DNNs and DL was also argued to be the state-of-the-art in domains like time series forecasting [God+21; Ale+20], for which various DNNs have been developed [Ran+18; Ore+20; Wan+19; Sal+20; Lim+21; Tür+21].

Lastly, similar to the modular approach of MLPs and DNNs, the *ensemble learning* approach aims at obtaining more expressive and adaptive models by training and combining multiple simple base models, i.e., $\boldsymbol{m} = (m_i)_{i=1}^{M}$ [Zho12; RN21]. For regression tasks, making predictions thus corresponds to averaging the prediction of all ensemble members, i.e., $\hat{y} = \frac{1}{M} \sum_{i=1}^{M} m_i(\boldsymbol{x})$, while classification usually commences via majority voting, $\hat{y} = \text{mode}(m_1(\boldsymbol{x}), \ldots m_M(\boldsymbol{x}))$. *Random Forest*s (RFs) for example represent ensembles built from decision trees, where individual trees are constructed from a bootstrap sample of the dataset at hand. The *eXtra Random Forest* (XRF) variant introduces additional randomness in constructing the individual trees, usually resulting in lower variance and higher bias [Ped+11]. With the *Adaptive Boosting* (AB) approach, ensembles are sequentially built, with each base model focusing on previously misclassified samples [Zho12].

This overview already shows the wide variety of methods for performing ML, each coming with individual advantages and drawbacks. In practice, many of the mentioned methods are not feasible for learning from raw data and instead require additional data pre-processing and feature engineering steps, which resulted in the idea of developing and deploying ML pipelines. As Section 2.2.1 will discuss, selecting methods and finding suitable models for

a new learning task and dataset is not trivial. Moreover, putting the introduced ML theory to practice requires some additional considerations.

2.1.4 Machine Learning in Practice

Unfortunately, the theory of ML introduced so far does not fully translate into practice. For a start, modern computers used for performing ML rely on floating-point arithmetic to represent real values $\mathbb{R}$, which might lead to unexpected phenomena like catastrophic cancellation [Mul+18]. Moreover, any ML model output $\hat{y} = m(\boldsymbol{x})$ depends on how the theoretical model is practically implemented, captured by the *execution environment* [Fis+22]:

Definition 2.4. *For putting ML into practice, the learning methods and algorithms are implemented via software and hardware, which together are defined as the execution environment $E \in \mathcal{E}$.*

As such, $\mathcal{E}$ is infinite and hypothetically contains all possible execution environments, while a single environment E represents a specific computational setup, mostly characterized by the installed hardware processor (e.g., a specific CPU or GPU), and the installed software library for performing ML. Various options for the latter exist, such as `TensorFlow` for training and deploying DNNs [Aba+16] or `Scikit-learn` as a toolkit for using various traditional ML methods [Ped+11]. It is important to acknowledge that the choice of environment can significantly impact the ML performance: Processors can differ in how they implement FLoating-point OPerations (FLOPs), different library versions might vary in their internal algorithmic logic, and software can always contain bugs and logic errors, from which ML libraries are no exception [Isl+19]. As a result, good scientific practice would necessitate to explicitly describe the execution environment used during ML experiments—unfortunately, these aspects are often under-reported, resulting in an observable "reproducibility crisis" [Hut18] and negative implications for SD that will be discussed in Section 2.3.2.

As the performance and predictions of ML methods and models are subject to the utilized environment, the same holds true for the model loss that is minimized during training, as introduced in Definition 2.3. In addition to the loss for individual samples or complete datasets, ML models moreover exhibit several other behavioral characteristics. First of all, there is a broad range of metrics for quantifying the *predictive quality* of a model based on given data from the task. Moreover, one might be interested in also assessing other practical aspects, like the *resource consumption* of training or predicting (also referred to as inference). As such, performance can be captured as practical model *properties* exhibited on a given *configuration* [Fis+24; FLM24]:

Definition 2.5. *Let D_T be a learning task dataset, $m \in \mathcal{M}_T$ be a corresponding model, and $E \in \mathcal{E}$ be the execution environment at hand. We first define $C = (D_T, E)$ as the investigated configuration, with $C \in (\mathcal{X}_T \times \mathcal{Y}_T) \times \mathcal{E}$. The model's practical behavior and performance can then be assessed via property functions $\boldsymbol{\mu} = (\mu_i)_{i \in \mathbb{P}_T}$, which quantify*

Table 2.1: Overview for ML Model Properties

Property	Group	Unit	Explanation
ACC1 (+)	Quality	Percent (%)	average top-1 ACCuracy
ACC5 (+)	Quality	Percent (%)	average top-5 ACCuracy
AR (+)	Quality	Percent (%)	Average Recall score
AP (+)	Quality	Percent (%)	Average Precision score
F1 (+)	Quality	Percent (%)	average F_1 score
mAP50 (+)	Quality	Percent (%)	intersection over union mean AP with 0.5 threshold
mAP95 (+)	Quality	Percent (%)	mAP with various thresholds between 0.50 and 0.95
MAE	Quality	Label-dependent	Mean Absolute Error
RMSE	Quality	Label-dependent	Root Mean Squared Error
MASE	Quality	Label-dependent	Mean Absolute Scaled Error
MAPE	Quality	Label-dependent	Mean Absolute Percentage Error
RTI	Resources	Seconds (s)	Running Time during Inference (per batch)
ENI	Resources	Watt seconds (Ws)	ENergy draw during Inference (per batch)
RTT	Resources	Seconds (s)	Running Time of Training
ENT	Resources	Watt seconds (Ws)	ENergy draw of Training
MP	Resources	Unitless	number of Model Parameters
FS	Resources	Bytes (B)	File Size of model on disk
FLOP	Resources	Unitless	FLoating-point OPeration

specific performance aspects of applying m on D_T using E. As such, the individual functions μ_i capture properties as positive real numbers, i.e., $\mu_i : \mathcal{M}_T \times (\mathcal{X}_T \times \mathcal{Y}_T \times \mathcal{E}) \to \mathbb{R}_{>0}$. The learning task T induces the applicable properties $\mathbb{P}_T \subseteq \mathbb{P}$.

To illustrate the concept of environments and model properties, we return to the example of classifying `ImageNet` images (D_T) with a model like $m =$ `MobileNetV2`, a DNN that is specialized for fast and resource-efficient inference [San+18]. Any practical implementation of this model will result in specific properties, like for example the required amount of time for classifying a single image, or the average *accuracy* of correctly classifying images. The property functions are used to describe the resulting model performance: For example, when deploying the pre-trained `TensorFlow` [Aba+16] variant on an *NVIDIA* `DGX A100` system (E), we observe $\mu_{\text{Time/Image}}(m, C) = 1.31$ milliseconds and $\mu_{\text{Accuracy (Top-1)}}(m, C) = 62\%$, meaning that the model on average correctly classifies three out of five images and can process about $\frac{1000}{1.31} = 763$ images per second [Fis+22]. Naturally, when evaluating other models or changing the choice of environment, the properties $\boldsymbol{\mu}(m, C)$ are expected to change—deploying the same model on a system without a GPU will for example likely increase the running time and could even have an impact on the model quality, due to possible differences in the processor logic. Specialized devices such as the *Google* Coral Tensor Processing Unit (TPU) [Cor24] theoretically allow for making the DNN deployment on standard hardware more resource efficient, however this claim is actually non-trivial to investigate [SFB24; Reu+19; Var+21]—a respective analysis will be conducted in Section 3.4.2.

An overview of relevant performance properties for this thesis is given in Table 2.1, grouped in terms of either describing the model's predictive quality or resource consumption. ACC1,

ACC5, AR, AP, and F1 are well-known quality metrics for classification [NZS23], mAP50 and mAP95 are common for image segmentation [JCQ23], MAE and RMSE are classic error metrics for regression, and MASE and MAPE are specialized for time series forecasting [HK06]. With respect to running time (also referred to as latency) and energy draw, the resource consumption can be assessed for the complete training (e.g., RTT) or single data batches (RTI), with the latter naturally being more relevant for model deployment. In addition to the resource properties, information like MP and FLOP can be used to capture the model complexity, which is naturally linked to the resource consumption. Similar to the `MobileNetV2` example, the later chapters use the properties from Table 2.1 as subscripts for μ, while μ_i or μ without a subscript is used to describe an arbitrary but fixed property.

It should already be obvious that model properties, while formalized as functions in Definition 2.5, are not trivial to evaluate and numerically quantify. Most of them for example require some data from the task, which however was deliberately not further formalized: Assessing MP and FS does not need any data, quality properties are commonly computed from the validation data, RTI and ENI just need arbitrary samples (the model output is irrelevant), and RTT and ENT naturally depend on the complete training data and process. While running time can be easily measured from code, the energy consumption is harder to quantify—as Section 2.3.3 will discuss, it is however possible to report reasonable estimates [Cou+24; Luc+19; Hen+20]. Note also that while a running time or accuracy of zero might be theoretically possible, this arguable should never be encountered in practical ML experiments. Definition 2.5 was therefore formalized to only consider strictly positive values as valid properties, which will be important for later formalizations. Moreover, smaller values correspond to better performance for most of the listed properties, however some require maximization for improvement, which is indicated by the $+$ in Table 2.1 and formalized via the constant σ_i [FLM24]:

Definition 2.6. *Let μ_i with $i \in \mathbb{P}$ be a specific property function. For each function, the constant $\sigma_i \in \{-1, 1\}$ indicates whether this property needs to be minimized $(\sigma_i = 1)$ or maximized $(\sigma_i = -1)$ for improvement.*

While not being of central importance for this thesis, it should also be mentioned that specialized quality metrics exist for assessing a model's prediction robustness. It can be increased by performing adversarial training, in which manipulated data instances, the so-called adversarial examples, are fed to the model in order to make it more robust against attacks on the input data [GMP18]. For example, several variants of the aforementioned `ImageNet` dataset were developed, allowing to assess model robustness toward common image corruptions and perturbations [HD19]. Progress in the field of adversarial training is for example captured by the respective *RobustBench* benchmark [Cro+21], which will be mentioned in Section 3.4.3. Similar benchmarks and competitive leaderboards exist for many other learning tasks and respective model properties, some of which will be discussed in Section 3.1.

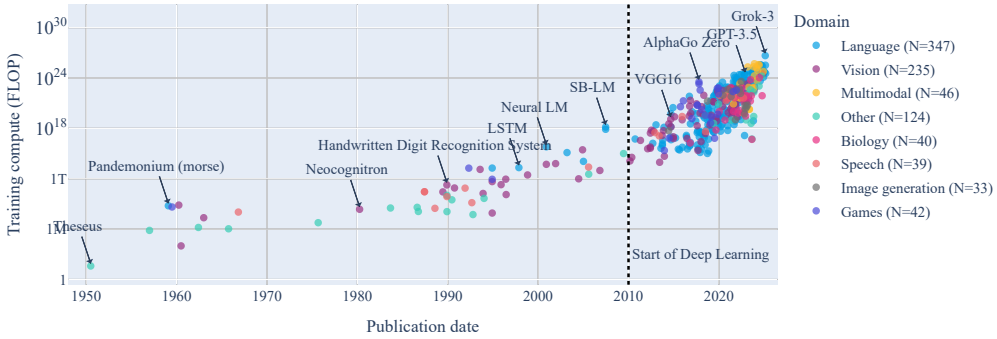

Figure 2.2: Trends of growing AI model sizes over the years, based on data by Sevilla et al. that clearly shows an upwards trend, particularly for DL [Sev+22].

In practice, model properties often correlate with each other, either positively or negatively, with the latter indicating trade-offs. An increased ACC1 for example will also positively affect the ACC5, however usually comes at the cost of higher model complexity, and thus, more parameters and a higher resource demand. Originally developed for multi-objective optimization [Yan14], the conceptual idea of *Pareto optimality* can be applied to balance different performance aspects [VLW25], as already shown in Figure 1.2. This understanding is crucial for advancing SD in ML and AI, for which Chapter 3 will introduce and apply methodology that allows to unify and compare model properties even across diverse environments.

2.1.5 Deep Learning and AI-as-a-Service

With the technical fundamentals of ML in mind, it is important to also shortly explain how the field moved toward the most recent AIaaS paradigm. Indeed, individual internet users and organizations today can access and use highly capable and complex AI models, and importantly, are not necessarily required to individually perform or even understand ML. This evolution was mostly fueled by advances in DL [LBH15], and more specifically, GenAI [Feu+24] and NLP [JM25; Bro+20], producing powerful and easy-to-use AI models and services.

As mentioned in Section 2.1.3, DL refers to the learning and deployment of DNNs, which are inspired by neuroscience [MP43] and represent modular networks of individual neurons [LBH15]. In the most basic MLP form [Ros58], DNNs consist of fully connected layers with neurons that non-linearly activate based on thresholding [RN21]. Early extensions to the basic idea of MLPs introduced convolution layers, residual blocks, and skip connections [LBH15], while the more recent attention mechanism [Vas+17] paved the way for transformer networks [Wol+20]. The variety of specialized neurons, layers, and modules, as well as the arbitrariness of arranging them, provides DL with a strong adaptivity. Whereas most other ML methods rely on extensive feature engineering when facing complex data, DNNs can thus instead learn latent feature representations in the

layers end-to-end [LBH15]. For image classification, DNNs can for example adaptively learn which image operations (e.g., convolutions) are useful for detecting objects in given data, whereas using other ML methods would require to first extract informative features via CV algorithms [Sze11].

The second advantage of DNNs over other ML approaches lies in their scalability. Due to almost exclusively relying on matrix multiplications, DNNs can be very efficiently trained and deployed on GPUs, which excel in parallel computing [LBH15]. The possibility to efficiently process large amounts of data, in combination with the high customizability and adaptability of DNNs, resulted in DL becoming state-of-the-art in learning domains like CV [Sze11], NLP [JM25], and time series forecasting [God+21]. However, the power of DL also resulted in a compute trend that Jensen Huang, chief executive officer of *NVIDIA*, recently referred to as a "hyper Moore's Law curve" [Hua24]. While the *Intel* co-founder Gordon Moore originally observed the number of transistors in CPUs to roughly double every two years [Moo98; Bro06], the compute in DL was instead observed to double every six months [Sev+22]. Figure 2.2, based on the data collected by Sevilla et al., displays the growing amount of FLOPs in ML models, clearly demonstrating how DL gave way to better but also bigger models[1].

With the concept of self-supervised learning [Bal+23], the power of DL was pushed even further. It primarily allows to also learn representations from vast amounts of unlabeled data, such as excerpts from the internet, thus reducing the effort of manual labeling. As model sizes can scale with the amount of available training data (recall Definition 2.3), self-supervised learning brought forth a new generation of NLP models that are commonly known as large language models [JM25], often consisting of billions of parameters [Bro+20]. They are able to generate textual answers for given text prompt and were also extended for the input and output of other data types, such as images and audio [Feu+24]. The versatility of these GenAI models allows to perform a variety of tasks, such as summarizing texts, describing images, recognizing speech, or generating videos [Luc+25]. Underscoring their "critically central yet incomplete character", such multi-task models are also referred to as foundation models [Bom+22]. While breaking the aforementioned Turing test [Bie23] and closing the gap to human reasoning [Cho+25], measuring the intelligence of foundation models is still an open question [KWI24] and many experts oppose the idea that current models match or surpass human intelligence [Alt+24; LS19]. Importantly, the remaining thesis will use GenAI as a central term for referring to large language and foundation models.

With billions of parameters, training GenAI models is extremely costly and often consumes several gigawatt hours of energy, and the inference energy cost can also quickly sum up to the equivalent demand of charging smartphones [LVL23]. As a result, GenAI models are commonly used in pretrained form and deployed in highly optimized remote execution environments (or in other words, datacenters). The cloud-based availability of AI functionality has fueled the paradigm of using AIaaS [Vad15], and in the times of multi-purpose GenAI models, this evolution can be seen as the fourth industrial revolution [ES20].

[1] Interactive exploration of this data is offered by *Epoch AI*, at `https://epoch.ai/blog/compute-trends`

The accessibility of AIaaS lowers barriers by allowing businesses to incorporate AI without acquiring specialized in-house expertise or on-premise compute infrastructure [Boa+18]. Many established companies and digital service providers like *Amazon*, *Meta*, *Google*, and *Microsoft* today also offer AIaaS to their customers. In addition to offering access to pre-trained models, multiple other forms of AIaaS exist, including local model deployment, fine-tuning, or infrastructure services [Lin+21]. This technological development also brought new global players onto the field: *OpenAI* is offering `ChatGPT` [Ope+24] as one of the most popular GenAI platforms, which at the end of 2024 served over 300 million weekly users and handled over one billion queries every single day [Rot24]. *Zhipu AI* provides similar services to the Chinese market [GLM+24] and several institutes have published custom GenAI models, including `Teuken-7B`, a language model developed by members of the *Lamarr Institute* [Ali+24]. In early 2025, the Chinese *DeepSeek* start-up "disrupted the proprietary dominance of Western AI", which demonstrated the instability of the market and re-strengthened the idea of open source AI [Sal+25]. To that end, *Hugging Face*, which originally became popular due to their extensive transformer library [Wol+20], transformed into an important company and community for developing, showcasing, and using open source AI [Hug24].

The availability of GenAI and AIaaS holds many promises, for example for democratizing healthcare [Tur+24] and advancing industry 5.0 [Sol24]. However, even the most renowned experts warn of "extreme AI risks" and highlight the importance of advancing research on AI safety and developing "adaptive governance mechanisms" [Ben+24]. A central problem is seen in bridging knowledge gaps and understanding the implications and trade-offs of using abstract AI services [Bre+23]. Moreover, the extreme resource consumption of GenAI [LJS24] and AIaaS [Cro24] needs to be critically viewed in the context of sustainability [LTM24; Sæt21]. Before this discussion is deepened in Section 2.3, we however first explore how well-performing AI models are practically (and automatically) developed in the first place.

2.2 Automation and Meta-Learning

Despite the availability of AIaaS, there are still reasons for performing ML and creating new models. When facing a domain-specific problem with respective learning data, a solution in the form of AIaaS has three prime issues: (1) While the extremely complex DNNs offered by AIaaS providers might also solve the problem at hand, they do not scale well, because of their high energy demand and resulting costs. (2) Multi-task AIaaS models do not necessarily solve the given task well, which would necessitate careful validation of any given output. (3) The complexity of DNNs, and foundation models in particular, result in high obscurity with regard to internal workings, so understanding how they solve the problem at hand (i.e., map inputs to outputs) is hard (if not, impossible). All these problems can be tackled by instead constructing a custom solution to the problem via traditional ML, which however is not straightforward in itself. This section therefore starts with formalizing the general problem of selecting promising models, then introduces

concepts for automating the model construction, and lastly discusses established AutoML frameworks.

2.2.1 Model Selection

For a specific learning problem and corresponding data, the problem of finding a good ML model can be defined as follows:

Definition 2.7. *Let $C = (D_T, E)$ be an investigated learning configuration and $\mathcal{M}_T$ be the space of applicable models. Model selection then corresponds to finding an optimal model $m^*_C \in \mathcal{M}_T$ with respect to some performance property $i \in \mathbb{P}_T$:*

$$m^*_C = \underset{m \in \mathcal{M}_T}{\arg\min}\, \mu_i(m, C)^{\sigma_i} \tag{2.4}$$

Note that obtaining $\mu_i(m, C)$ includes the secondary optimization problem of training the respective model on the available data D_T, as given in Definition 2.3. Definition 2.7 describes the most basic and well-established understanding of model selection, where the best-performing model with regard to one specific (quality) aspect should be chosen—Section 5.1 will later introduce a more sophisticated approach. Generally, it is accepted that for the optimization problem of model selection, "no such thing as a free lunch" exists [WM97]. Commonly referred to as the No Free Lunch (NFL) theorem, this indicates that ML methods behave rather unpredictably for different tasks, and as a result, there are no easy shortcuts for finding good models. Equation (2.4) thus is usually approached by testing candidate models for the configuration at hand, using (cross-)validation techniques to avoid overfitting [HTF09].

It is important to better characterize the search space of applicable models $\mathcal{M}_T$ along several interrelated aspects. First, one can select among various ML methods, second, additional preprocessing or feature engineering on the raw data is often required, and thirdly, most ML methods can be customized via so-called hyperparameters (as already mentioned in Section 2.1.2). Universal examples for hyperparameters are regularization terms, which are added to the loss function, or configuration options for controlling the training, such as learning rate, batch size, and trainable parameter initialization. For complex and modular models, the choice of ensemble composition or DNN architecture can also be understood as a hyperparameter, which even induces the number of trainable model parameters as well as how they interact with each other. Tuning only the hyperparameters is commonly referred to as Hyperparameter Optimization (HPO). Naturally, method selection, data preprocessing, and HPO are also subject to "conditionality" [HKV19], and thus sometimes even addressed at the same time, which is commonly referred to as full model selection [EMS09] or combined algorithm selection and HPO (CASH) [Feu+15].

Instead of searching the complete space $\mathcal{M}_T$, one could also consider to search in a smaller subspace, or even limit the search to a pre-defined finite set of candidates, with individually fixed preprocessing and hyperparameters. As an example, consider facing a novel time

series forecasting dataset, for which a good model needs to be found. For this domain, a large assortment of DNNs have already been proposed [Ale+20], so instead of searching for new networks, one could resort to select and train the best model from this pool—this application scenario is discussed in full depth in Section 5.2. Model selection from a finite set of candidates can either be performed on a model pool with pretrained classifiers, or in terms of choosing models to train in the first place. Closely connected is the problem of ensemble pruning [TPV09; Zho12], where the size of an ensemble is reduced by discarding single members (i.e., *selecting* which members to keep), which sometimes even increases the ensemble quality.

ML experts draw from their individual experience when navigating model search spaces manually, or in other words, they perform ML engineering by choosing and evaluating applicable methods with respective hyperparameters. But in order to streamline the creation of ML solutions and reduce manual effort, researchers and developers have also put a lot of work into *automating* the intricate process of ML engineering, commonly referred to as AutoML.

2.2.2 Automated Machine Learning

Hutter et al. frame AutoML to be "a democratization of ML" [HKV19], as it makes customized state-of-the-art solutions available to application domain experts with less ML expertise. Modern AutoML frameworks were indeed demonstrated to rival and in some cases even outperform human experts that manually engineer the problem, for example in the context of ML competitions [Eri+20]. As a result, AutoML has become omnipresent for training new models, for example in the form of Neural Architecture Search (NAS) methods that allow to construct new DNNs [WRP19]. In the following, the more general idea of automated HPO, the special case of NAS, and the fundamental concept of meta-learning are introduced.

Hyperparameter Optimization

The intelligent navigation of ML hyperparameter search spaces has been explored for over thirty years [KJ95] and is still a central problem in modern AutoML [FH19]. Traditionally, HPO can be approached via a grid search (also known as parameter sweep) or random search, with the latter empirically being more efficient [BB12]. While suffering from the curse of dimensionality, performing these searches can at least be easily parallelized, because points in the search space (i.e., different hyperparameter settings) are usually independent from each other. Other approaches use Bayesian optimization [Sha+16] and Gaussian processes to predict how models might perform when trained with specific hyperparameters [SLA12]. In practice, this results in less evaluations than a random search, because the probabilistic model has a chance to identify and prevent redundant runs. For some ML methods it is possible to compute gradients with respect to the hyperparameters, allowing for even more efficient HPO approaches [Lar+96]. Because testing a single

hyperparameter setting (i.e., training and evaluating the respective model) can be expensive for complex problems and data, many approaches focus on multi-fidelity optimization, in which the candidates' training loss function is approximated with a substitute function of lower fidelity [FH19] (e.g., via learning curve extrapolation [DSH15]). HPO is still actively researched, with the search for standardized benchmarks, scalability to large-scale problems, and strategies to combat overfitting being the most pressing issues [FH19].

Neural Architecture Search

NAS is an important subfield of AutoML and focuses on automating the design of DNN architectures for reducing the reliance on human expertise. With the rise of DL, first approaches used RL to identify good architectures [ZL17; Bak+17]—today, it is common practice to spend thousands of GPU-days for finding well-performing DNNs [LZJ22]. Advances for NAS are typically categorized along three dimensions [EMH19a]: the search space, which defines the potential architectures to explore; the search strategy, which dictates how to navigate this space; and the performance estimation strategy, which assesses the efficacy of the candidate architectures. Search spaces can be grouped into either functioning globally, cell-based, or based on recurrent cells [WRP19]. As in general HPO, diverse paradigms such as Bayesian optimization, evolutionary methods, RL, and gradient-based methods are used as search strategies [EMH19b]. As DNNs are especially costly to evaluate and train, methods for multi-fidelity optimization are centrally used as performance estimation strategies in NAS. With hundreds of published NAS papers, the high computational demands and the resulting need for efficient search algorithms remain the most active areas of research [LZJ22; Ben+21].

Learning to Learn

By now, it should have become clear that large parts of AutoML research builds on the conceptual idea of solving additional meta-learning tasks. Also known as the problem of "learning how to learn" [Sch87], this for example allows to learn from prior experience or predict suitable next candidates. Meta-learning enables faster adaptation and generalization to new tasks and therefore has strong connections to multi-task and transfer learning, which are of central importance in modern DL [Vet+24]. Performing meta-learning requires some sort of meta-data about the task or problem at hand, which can take various forms. In AutoML, this data usually describes method configurations (e.g., hyperparameters) as well as the resulting model's performance properties [Van19; Van+14], such that meta-learners can for example be trained to predict the expected performance of a candidate. In other scenarios, one could use data about the task at hand, which are commonly referred to as meta-features. By manually deriving statistics over the data [WSS16] or using a learned meta-feature extractor like `Dataset2Vec` [JSG21], one can not only assess the similarity of different ML tasks, but also meta-learn to predict the performance of different method configurations without testing them on the specific data. Lastly, meta-data

can be derived from trained models, i.e., their hyperparameters and trained parameter weights. This is for example highly relevant for few-shot learning [Vet+24], in which accurate models are trained based on only a few training instances [Lak+17].

2.2.3 Established Frameworks

One of the first AutoML frameworks is `Auto-WEKA`, which simultaneously selects learning algorithms and tunes their hyperparameters [Tho+13] on base of the `WEKA` ML software toolkit [Hal+09]. For most other established ML and DL libraries, corresponding AutoML extensions have been published, such as `Auto-Sklearn` [Feu+22], `AutoKeras` [JSH19], and `Auto-PyTorch` [ZLH21]. The first focuses on creating ensembles of classical ML methods for tabular data, whereas the others allow for automatically constructing DNNs and also consider more complex data domains like images, texts, or time series [SJH22]. Another popular framework is `AutoGluon` [Eri+20], which automatically constructs a single ML ensemble by greedily adding base learners if they improve the predictive performance on the data. A respective extension for time series forecasting is also available [Shc+23].

The `Naive AutoML` [MW23] approach subdivides the complete search problem into sub–problems and addresses each of them independently, which in practical comparison with other AutoML systems often results in competitive qualitative performance while being more cost efficient [MW23]. As a totally different approach to AutoML, the `TabPFN` approach uses a prior-fitted transformer DNN [Hol+23] that can produce predictions on tabular data with a single forward pass. Due to only priming the network instead of requiring actual learning of the train data, `TabPFN` requires much fewer resources than other AutoML frameworks, however it is limited in terms of supported dataset sizes and rather obscure in terms of its internal model complexity. For some specialized domains and tasks like time series forecasting, there exist further extensions or specialized approaches to AutoML [Als+22; LD22; Ale+20; SJM22].

In their valuable benchmark of various AutoML frameworks [Gij+24], Gijsbers et al. conclude that the average rankings in terms of predictive capabilities are generally competitive, however `AutoGluon` consistently scores best. At the same time, however, some of the tested frameworks do not scale well with dataset sizes and many of the obtained models are very costly to evaluate, which requires to perform a more resource-aware analysis. This is in line with the call for “green” and “transparent” AutoML, with proposed practices like adding explicit sustainability checklists to papers [Tor+23]. When also assessing and comparing resource consumption, there is no clear winner in AutoML, as there are obvious performance trade-offs between predictive quality and the complexity of both the search as well as the ultimately obtained model [NLA25].

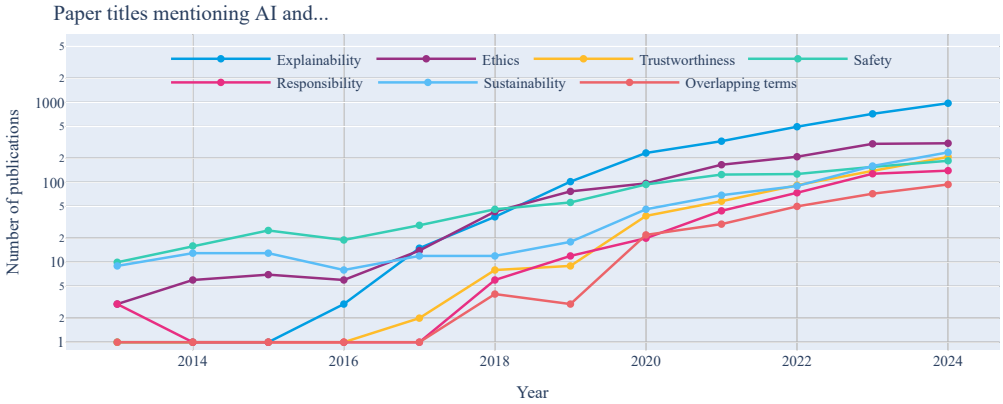

Figure 2.3: Number of peer-reviewed publications that explicitly thematize the different research subfields that can be unified under AI sustainability.

2.3 AI and Sustainability

The rapid evolution of ML methods and AI systems, while providing many benefits for our world, is also critically discussed in terms of consequences and dangers for our society, economy, and environment, representing the three dimensions of sustainability. As already highlighted with Figure 1.1, this thesis understands sustainability as an umbrella term for the various overlapping research topics and fields that focus on the risks and implications of AI advancements. The first part of this section introduces these fields, while Section 2.3.2 discusses how they can be connected by the idea of SD. As this thesis focuses on practical aspects of AI sustainability, the background closes with some fundamental considerations as to how the sustainability of specific ML methods and AI models can be quantified, with regard to environmental impact and other factors.

2.3.1 Adjacent Research Fields

Due to the transformative power of AI technology, several research communities analyze the potential harms of AI systems and develop means for reducing them. As mentioned in Section 1.1, this includes research on explainable [Sam+19; Lan+21], ethical [Flo+18], trustworthy [Cha+21], safe [Ben+25], and responsible [Dig19] AI, among several other directions. As a quantitative overview, Figure 2.3 depicts the yearly number of corresponding scientific publications, obtained from a straightforward keyword search on titles in the `DBLP` database [Ley02], with a focus on peer-reviewed AI literature[2]. Noting the log scale on the y-axis, an impressive growth of all fields can be observed across the years, with hundreds of yearly publications since 2020 and XAI most likely surpassing 1000 publications in 2025. The following gives an overview of the fields and discusses

[2]The script for this search can be found at `https://github.com/raphischer/strep`

their connections—keep however in mind, that cleanly separating the fields arguably is impossible, which can also be seen from the noteworthy overlap curve in Figure 2.3.

According to the search, XAI research currently is the most active field and discusses how models and their internal workings can be be made more transparent, in order to satisfy user needs and adequately fulfill their "right to explanation" [SP18]. Many basic ML methods produce models that are considered to be inherently interpretable [Rud19], due to accompanying predictions with information like feature importance [SJ21]. However, the question of explainability is much harder to answer for complex models like DNNs [Sam+19]. A wide range of XAI methods was proposed to make these "black-box" models more transparent [AB18], either with regard to individual decisions, or in terms of the overall model logic (commonly referred to as local and global explainability [Hol+22]). Prior knowledge about the learning task at hand [Rue+23] is often infused into XAI, and obtained explanations are frequently used to iteratively refine and improve AI systems [Bec+23]. Several works have attempted to build useful taxonomies of proposed XAI approaches [Ang+21; Bar+20] and guide users with selecting appropriate XAI methods [Spe22]. A central problem in XAI lies in the evaluation of explanations due to missing ground-truth data [Nau+23], and moreover, it is questionable whether there even exists a single method that satisfies all needs [SF20]. As a result, the XAI field of today is extremely broad—it not only comprises a large corpus of intricate methods for obtaining explanations, but also a lot of interdisciplinary literature [Lon+24], which for example aims at "identifying and clarifying desiderata of the various classes of stakeholders" [Lan+21].

XAI is also of central importance for the other research areas listed in Figure 2.3. It was argued that explainability "can be utilized to ensure fairness, robustness, privacy, security, and transparency", which are considered to be the six fundamental "pillars" of responsible AI [BX23]. The idea of AI responsibility is naturally linked to ethical considerations [JIV19] about establishing a "good AI society" [Flo+18] and hence addressing them requires a good "understanding of socio-technical interactions" [Dig19]. The closely related conceptual idea of human-centered AI has been discussed for many years [CB23; Shn20] and today plays a central role for industry 5.0 [MBC24]. Establishing fair [Meh+21; Kus+17], ethical, and responsible AI systems also goes hand in hand with advancing their trustworthiness [Cha+21; Bru+20; Cre+19] and making them accountable for their decisions [NTF24; HKZ23]. From an interdisciplinary perspective, trust in any technology is fundamental for the general uptake and appropriate usage of systems [LS04; MW07]. Hence, it is of utmost importance to calibrate users' trust appropriately to the actual trustworthiness of any automated or intelligent system [WKM23]. One has to differentiate between "different types of trust" [Mck+11], for example a duality of trust can be observed for cloud services, relating to trust in service providers and trust in Information Technology (IT) artifacts [LS16]. The trustworthy economic use of AI thus requires a "harmonized interplay between product and organizational perspectives", which could for example be established by systematic quality assurance [Sch+22] or mechanisms such as red team exercises and audit trails [Avi+21]. The latter links back to responsibility, as an interdisciplinary study proposed audit areas that closely resemble the aforementioned pillars [Cre+19], and could also be utilized for sustainable AI development [GM22]. It should also be noted that several

works highlighted the need for trustworthy reporting [Geb+21; Mit+19; Arn+19], which will be more centrally discussed in Chapter 3.

Naturally, the research across all these fields is also highly relevant for AI regulatories. In that context, the EU is seen as having a pioneering role, anticipating a "so-called Brussels effect" that also followed the introduction of the *General Data Protection Regulation* in 2016 [Mök+22]. Few years later, the EU announced to work on the "world's first rules on AI", demanding intelligent systems to be designed "safe, transparent, traceable, non-discriminatory, and environmentally friendly" [Eur23]. After many iterations, the final version of the *EU AI Act* came into force in August 2024, prominently promoting "the uptake of human-centric and trustworthy AI" [PE24]. Comparable legislative approaches also exist outside of the EU, for example in form of the United States of America *Algorithmic Accountability Act* [Uni22], demanding businesses to report about any utilized automated decision systems and investigate their impact on consumers. In comparison to the *EU AI Act*, it was however argued to be "too modest", constituting "only a fragmented attempt" at regulation that requires additional institutional backing [Mök+22]. Shortly later, Joe Biden (46th president of the United States of America) signed an AI executive order [Bid23], which prominently highlights the importance of safety, security, and trustworthiness, focusing for example on establishing standards for critical infrastructure, driving AI-enhanced cybersecurity, and applying consumer protection laws to AI. In response, several States have individually approached the regulation of AI, such as the California initiative that explicitly fosters a "safe and responsible innovation ecosystem" in the context of GenAI [New23]. The *ELVIS* act in Tennessee focuses on the protective rights of artists, specifically aiming to regulate deepfakes with regard to image, voice, and likeness [Sta24]. Utah also published a respective policy act, enforcing transparency to protect consumers from potential GenAI harms [Uta24]. Recently, the Biden executive order was repealed by Donald Trump, the following president, who aims at deregulating AI in order to drive innovation over safeguarding—this new direction is viewed critically, especially in the context of climate change and public health [WK25].

2.3.2 Sustainable AI Development

While sustainability receives slightly less attention in AI literature and regulatory approaches, it is here used as an umbrella term for the various aforementioned research fields (as already visually depicted in Figure 1.1). Note that this is not a completely novel take—many works have connected sustainability to AI transparency and explainability [Cha24; Roh+24; Tru20], ethics [GM22; Wyn21; JIV19], responsibility [Roh+24; GM22; Tru20], and trustworthiness [GM22; Tru20]. A comparable research field analysis and diagram was contributed by Khakurel et al. [Kha+18], and according to Genovesi and Mönig, sustainability is of central importance for the "ethical certification" of AI [GM22].

The United Nations *Agenda 2030* arguably is the most fundamental resource for discussing SD, balancing economical, societal, and environmental needs. In a global commitment, 193 member states agreed upon striving for 17 SDGs, which are accompanied by several

hundred targets and indicators for tracking progress [Col15]. The latest report reveals mixed outcomes in terms of advancing the SDGs: despite certain health improvements the global health progress has slowed, education remains under threat, conflict-driven displacement and casualties are at record highs, and urgent actions are needed for resolving conflicts and accelerating climate efforts [EA24]. Closely connected to the SDGs are the well-known ESG criteria, which allow for tracking corporate performance with regard to Environmental, Social, and Governance aspects [RDI23], thus operationalizing the principles of sustainability. In the context of SD, AI was acknowledged to play a two-fold role [Wyn21]: On the one hand, it can be used as a "vehicle to meet the SDGs" [Di +20], however at the same time, it is also important to investigate the sustainability of AI itself.

AI for Sustainability

Several projects like *AI for Good*[3] and the *International Research Centre on AI under the auspices of UNESCO*[4] successfully drive SD with the help of AI, for example to better manage natural disasters [Int24b]. In a literature review by Di Vaio et al., evidence for sustainability benefits in terms of business models as well as production and consumption patterns was presented [Di +20]. Shortly later, Kar et al. surveyed over 280 papers describing how AI is successfully used for advancing SD in various domains like construction, transportation, healthcare, and manufacturing [KCS22].

With these seemingly positive results, it is however important to keep in mind that reports are often biased and might even be subject to greenwashing [MT23], which for example was discussed in a review of SD in the context of smart cities [Sha+24]. To give another example, an early consensus-based elicitation study by Vinuesa et al. suggested that "AI can enable the accomplishment of 134 targets [...but] may also inhibit 59 targets" [Vin+20]—however later, this work was criticized for overstating the positive AI impact and disregarding potential negative consequences [Sæt21]. Concisely answering whether AI truly benefits sustainability is especially difficult because of ambivalence, for which Rohde et al. gave a good example: Using AI for remote sensing could serve positive and negative use cases (with regard to sustainability), as it could for example advance "agricultural productivity" but also "gas exploration" [Roh+24]. The current wave of AI ethics is expected to "place SD at its core", with a special focus on addressing "the sustainability of developing and using AI systems in and of themselves" [Wyn21].

Sustainability of AI

With the sustainability dimensions in mind, negative consequences of AI systems should be acknowledged with regard to implications for the society, economy, and environment [Hal22]. In their broad study, Khakurel et al. for example discuss how AI might

[3] https://aiforgood.itu.int/
[4] https://ircai.org/

potentially "displace low-skilled workers", cause "disruption" in the IT industry, and "affect interactions or social isolation" [Kha+18]. The increasing availability of AI, as discussed in Section 2.1.5, is anticipated to "having detrimental impacts on future generations" that could lead to "intergenerational injustice" [Hal22]. Critically, "surveillance capitalism is on the verge of dominating the social order" [Zub18], meaning that currently, the power of GenAI and AIaaS technology resides with few big companies, which might not sufficiently acknowledge the importance of sustainability [Sæt21; VLW25].

While analyzing the impact of AI for the society and economy is important, it was argued that the "environmental disaster of our time" should receive the most urgent attention and action [Wyn21]. Environmental sustainability is hindered by the fact that values in ML research are "operationalized in ways that disfavor societal needs" [Bir+22], which naturally results in the observable compute trends [Sev+22] and "bigger-is-better" paradigm [VLW25] (as already shown in Figure 2.2). Despite many works that have investigated the resource consumption and efficiency of modern ML for domains like CV [Gar+19; Sch+20], stream mining [Gar+19], and NLP [LVL23; Wu+22; SGM20], the AI field is still "overly focusing scale" resulting in "economic inequalities and environmental (un)sustainability" [VLW25].

To counter the current trend and make AI more sustainable, various practices and workflows have been proposed. Fostering resource efficiency and inclusivity, Schwartz et al. suggested to categorize approaches and models into Red AI and Green AI, with the latter explicitly "taking into account the computational cost" of advancing the state-of-the-art [Sch+20]. While the term "green" is for example also used to encourage SD in the AutoML community [Tor+23], a large-scale categorization of approaches and models has not yet been established. In that context, measuring or estimating the environmental impacts of ML is crucial—several works have approached this problem [Gar+19; Luc+19; Hen+20; Bud+22; Cou+24] and will be further discussed in Section 2.3.3. The resulting methods and tools were used to evaluate the environmental cost of training or running AI models, for example focusing on the NLP domain [SGM20] or GenAI model repositories like *Hugging Face* [Cas+23; LJS24]. A recent study unveiled that the environmental impact of using publicly available GenAI models quickly compares to fully charging a smartphone, and that multi-purpose models and image tasks are especially energy-intensive [LJS24]. While the big AIaaS providers often disclose exact numbers, they process billions of GenAI queries per day [Rot24] and recently even started to reopen nuclear power plants to meet their electricity demands [Cro24]. According to reports by *Google* [Pat+22] and *Facebook AI* [Wu+22], inference makes up about 60% of the total ML energy use, however these numbers might have changed since 2022 (which was around the time when GenAI and AIaaS practically took off).

As a step toward environmental transparency, a recent initiative aims at accompanying pre-trained AI models with a comprehensive energy score rating [Luc+25]—as Section 4.1.4 will discuss, this closely relates to my own work on the topic [Fis+22]. In combination with explicitly considering and advancing ML efficiency during model deployment, Patterson et al. believe that improved datacenter transparency (e.g., communicating the power

usage effectiveness) will eventually result in a shrinking ML carbon footprint [Pat+22]. Other voices are more pessimistic, arguing that the growing environmental costs of AI represent an example for *Jevons' paradox*, an effect that was originally observed in economics [Jev65; Alc05], where increasing the efficiency of a resource ultimately does not lower the demand, but instead rebounds to a rising total consumption[5]. Building on the realization that increasing AI efficiency is not guaranteed to lower the overall resource costs, the authors further argue that "climate-aligned AI strategies might require public policy frameworks" [LSC25]. As such, it is imperative to adopt "a variety of technical, behavioral and organizational interventions" [LTM24], in combination with reinterpreting existing legislation and establishing novel "policy measures to align AI and technology regulation with environmental sustainability" [Hac24].

2.3.3 Quantifying Sustainability

Investigating the environmental costs of GenAI models, Luccioni et al. conclude that practitioners need to "practice transparency regarding the nature and impacts of their models" [LJS24], which naturally requires means for quantifying and assessing respective factors. In the general context of sustainability and SDGs, progress is commonly tracked via associated targets and indicators. Rohde et al. contributed a corresponding AI assessment framework, which features criteria and indicators for "the conscious development and application of AI systems" [Roh+24]. While this work does an excellent job at considering all aspects of SD and linking works relating to the individual indicators, it unfortunately does not adequately address as to how they can be measured in practice. Self-assessment via questionnaires, as proposed by the authors, is naturally prone to ambivalence and human misjudgment, and moreover does not scale well. As computer scientists, we rather look for measurable model properties like the ones mentioned in Section 2.1.4, and the following will accordingly explore how aspects of sustainability can be quantified.

Environmental Impact and Energy Consumption

With the established importance of environmental sustainability [Wyn21], respective quantifications should investigate greenhouse gas emissions and global warming effects caused by developing and using AI models. For simplification, the standardized measure of CO_2-equivalents was developed, for example describing the carbon intensity of transportation (grams of CO_2-equivalent per person-km) or energy (grams of CO_2-equivalent per kilowatt hour). A practical approach toward estimating CO_2-equivalents for ML is available in the form of the `ML Impact Calculator` [Luc+19]. It allows users to specify the key characteristics of their ML experiments, from which the total amount of CO_2-equivalents is simply estimated via:

[5] Named after 19th century economist William Stanley Jevons, who analyzed that the increased efficiency of coal-burning resulted in much higher demands, fueling the first industrial revolution [Jev65].

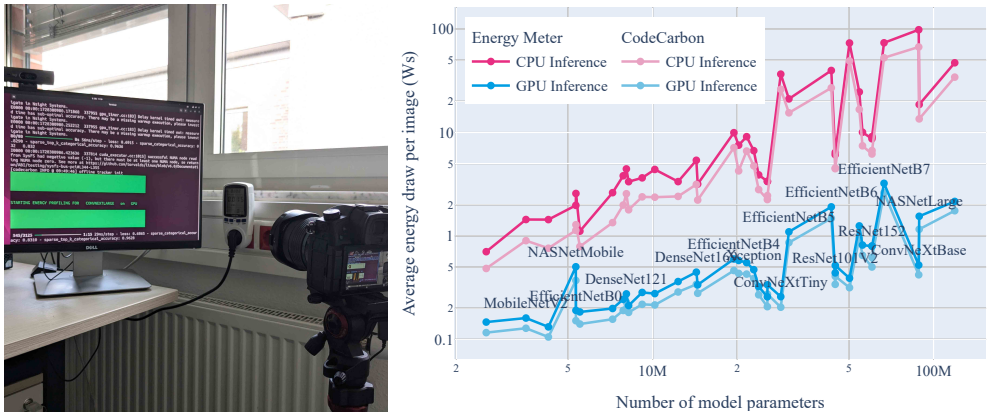

Figure 2.4: Differences of assessing energy draw of inference with ML models via the `CodeCarbon` software or an external energy meter. Experimental setup is shown on the left, more information can be found at `https://github.com/raphischer/ai-efficiency-testbed`.

$$\text{CO}_2\text{-equivalents} = \text{Power consumption} \times \text{Time} \times \text{Carbon efficiency} \tag{2.5}$$

While the tool offers guidance for estimating the impact of individual calculations, one has to acknowledge that a holistic assessment would require to consider the complete AI life cycle [Wu+22]: The environmental impact is thus subject to (a) the specific task to solve (e.g., training a model, fine-tuning a model, inference with a pre-trained model, see also Section 2.1.1), (b) the energy consumption of the utilized execution environment (e.g., a local desktop, optional add-on acceleration hardware, a remote compute node, a federated compute datacenter, as explained in Section 2.1.4), and (c) the corresponding carbon efficiency (which mostly stems from the energy mix of the local power grid, but could for example also be influenced by additional green energy sources). Several secondary factors, such as carbon offsetting, the procedure of assembling training data, or the embodied carbon in utilized hardware and infrastructure, further complicate the accurate assessment of total emissions and impact, and hence are often neglected or only partially assessed [LTM24].

While the `ML Impact Calculator` [Luc+19] is a helpful tool for obtaining a rough estimate, it fails to acknowledge that the power consumption of the hardware at hand is not a static constant, but actually strongly depends on the executed ML experiment. For this reason, several tools have been developed that allow for directly profiling the energy consumption of the execution environment during the experiment. With a straightforward drop-in approach, the `experiment-impact-tracker` allows to track the energy consumption of the most important hardware components, namely the CPU and GPU [Hen+20]. The later published `CodeCarbon` software follows a similar approach, however is still actively maintained and also includes a visual dashboard for tracking resource use and emissions [Cou+24]. Under the hood, both tools estimate the overall energy consumption

and environmental impact via internal hardware profiling, using tools like *Intel*'s running average power limit [Int24a] and *NVIDIA*'s management library [NVI12].

While allowing to track the environmental impact of ML and AI, resource-aware ML engineers should keep in mind that these tools currently underestimate the energy consumption, as certain hardware components are hard to profile (e.g., USB sockets, network adapters, hard disks)—as such, using external energy meters allows for more precise measurements. During writing this thesis, additional experiments were performed for comparing `CodeCarbon` results with external measurements, as displayed in Figure 2.4. It shows that for inference with pretrained image classifiers of various sizes, `CodeCarbon` is indeed able to closely approximate the real power consumption of the system—both for CPU and GPU inference. Noticing the log-scale, one can however also observe that differences grow with the model size, and that the energy measurement difference is not the same for CPU and GPU inference—while not yet scientifically published, these first results can already be explored in more depth at `https://github.com/raphischer/ai-efficiency-testbed`.

It should also be noted that specialized hardware like USB accelerators [Cor24; Int22] or *NVIDIA* `Jetson` boards are often not natively supported by the aforementioned profiling libraries and require customized solutions [SFB24; Fis+24]. Moreover, these libraries are tricky or even infeasible to use on shared machines, federated compute clusters, and cloud-based computing, which are commonly used for deploying GenAI models. Unfortunately, to the best of my knowledge, AIaaS providers currently do not openly communicate the exact environmental costs of using their services. However, going back to Equation (2.5), the information on used hardware (i.e., GPU type), running time of experiments, and datacenter location allows for rough approximations. For the later discussed experimental results in this thesis, energy consumption and the ENI and ENT properties from Table 2.1 were generally measured via `CodeCarbon`, if not otherwise specified. To foster transparent reporting, most of my later papers provided estimates for the amount of carbon emissions caused by executing the respective experiments. Based on this information, I estimate the total emissions caused by assembling the contents for this thesis to 200 CO_2-equivalents—about double the amount of the most resource-hungry experimental investigations [FS24; Fis+24; FLM24], thus including a fair overhead for smaller experiments and the writing process.

Quantifying other Sustainability Aspects

In addition to environmental aspects, measurable properties for quantifying AI sustainability with respect to the other dimensions are needed. On a business or service provider level, audit trails can be used for assessing how sustainable the development of AI models is [GM22; Avi+21; Cre+19]. Similarly, Rohde et al. and Rutinowski et al. rely on expert-assessments for establishing sustainability criteria [Roh+24] and benchmarking trust [Rut+24], which however do not scale well due to high manual effort and bias potential.

Few works have discussed how sustainability criteria with regard to society and economy can be quantified for individual models, and ideally without human feedback. With a unified and widely adopted way of documenting datasets [Geb+21], a score could possibly be derived for describing the quality of the data used during model training—however, such a documentation format is not yet available or established for all datasets. For the aspect of transparency, the number of model parameters as well as some general information on the model functionality (e.g., number and types of DNN layers) can be used as indicators [Roh+24]. Further works suggested methods for quantifying model explainability, for example based on the heights of model decision (i.e., explanation) trees [ZSW22]. The concept of neuron coverage [Pei+19] can be used to assess which parts of a DNN is activated by single inputs, hence allowing to quantify how well a larger set of test inputs covers the model at hand [ZSW22]. Regarding fairness, the causal approach proposed by Kusner et al. can be used to test whether models are fair [Kus+17], and Bordia et al. introduced a method for quantifying biases in NLP models [BB19]. Similarly to explainability, fairness assessments however require that protected attributes or biases in the data are known a priori (i.e., as ground-truth), for testing models against them. Lastly, certain quantifiable usability indicators such as number of scientific citations, code availability, and user scoring (for example, based on `GitHub` stars) could be assessed.

Concluding this chapter, the relations between ML, AI, and AutoML should have become clear, as well as the connection to the importance of establishing more sustainability in the field. Based on these fundamentals, the next chapter proposes and evaluates methods for better reporting on AI advances and properties. The respective methodology benefits transparency and aids decision making in order to facilitate sustainable AI development.

3 Sustainable and Trustworthy Reporting

Having covered the necessary fundamentals, we now explore the first central contribution of this thesis, namely the STREP methods and software for reporting on AI advances in a sustainable and trustworthy way. This framework was developed for facilitating SD by addressing the need for overcoming knowledge and communication gaps, which for example occur between domain experts, who thoroughly understand the learning problem at hand, and ML engineers, who are supposed to develop intelligent solutions. As a personal example for real-world communication gaps, Section 1.1 discussed my experience of working for the industrial manufacturer *Wilo*, where we explored AI use cases that involved various stakeholders within the company [Fis+23]. Based on the experience from working in multidisciplinary teams, Piorkowski et al. analogously highlighted that bridging such gaps requires customized documentation (i.e., reporting) that is tailored for the target audience [Pio+21]. In addition, the importance of good education and AI literacy was emphasized in a much broader meta-summary of experiences described by nearly 5000 ML practitioners [Nah+23], yet concisely defining this literacy and establishing guidelines for improving it is not straightforward [Ng+21]. While AIaaS reduces the need for employing ML experts in order to benefit from AI, non-experts will struggle to understand the practical properties and trade-offs [Bre+23] (see also Section 2.1.5).

Obviously, knowledge and communication gaps can result in questionable decisions, which is problematic in the established context of sustainability and trustworthiness (Section 2.3). Greater transparency and a better understanding of practical AI implications are needed, however improving reporting requires to first thoroughly characterize the current state and identify associated problems, which will be the focus of Section 3.1. Afterwards, Section 3.2 will present the STREP framework as a means for establishing better reporting based on a five-step methodology [FLM24]. These theoretic formulations are complemented by a practically implemented software suite, which will be explored in Section 3.3. The last section will demonstrate the practicability of STREP by reporting on the performance of selected models in sustainable ways [Fis+22], analyzing the impact of using different computing environments for model deployment [SFB24], and lastly uncovering biases in established report databases [FLM24]. The concepts introduced in this chapter act as a fundament for the later chapters of this thesis.

3.1 The Current State of Reporting

Realizing that better reporting is vital for bridging knowledge and communication gaps, we first have to understand how reporting currently takes place. The following presents a respective characterization that is inspired by the original work on STREP [FLM24], after which benefits and shortcomings are discussed.

One can differentiate between six different types of reporting in the context of AI advances. The **first** and most credible format for reporting new AI insights or presenting new models arguably are scientific *papers*, i.e., peer-reviewed publications that describe new models and results in methodical depth. Additional *gray literature* like preprints, blogs, social media posts, library documentations, videos, and journalist publications represent the **second** type of reporting, which naturally became more relevant and frequent thanks to the growing accessibility of and interest in AI. These reports often build on scientific publications and break down their contents to inform a broader audience, while discarding much of the technical depth. The **third** reporting format refers to the more user-oriented *model summaries*, like the today well-established *model cards*, which are conceptualized to "clarify the intended use cases" as well as encourage "transparent model reporting" [Mit+19]. While originally proposed by *Google*, this idea is also successfully implemented by *Hugging Face* [Hug24] and *IBM* [Arn+19], with the latter even having patented their respectively named AI *fact sheets* [Arn+22]. The initiative to establish concise "datasheets for datasets" [Geb+21] also needs to be mentioned in this context, as an early pioneering project for standardizing the reporting on datasets that was adopted by several researchers and companies.

The idea of established consumer-oriented systems like the EU textile care labels [Eur24b], energy labels [Eur24a], or Nutri-scoring [Org22] fueled the concept of even more abstract *AI labels*, representing the **fourth** type of reporting [SSW19; Fis+22; Luc+25]. By hiding away the intricacies of ML, these reports are designated to only inform on the most important practical aspects of AI models. Labeling is one of the central contributions discussed in this thesis [Mor+22; Mor+21; Fis+25] and will be the focus of Chapter 4. Moving away from individual results, the idea of statistically comparing multiple models has a long history [Dem06] and paved the way for the **fifth** type of reporting. Today, one can indeed find many competitive *benchmarks* and resulting leaderboards, which summarize the performance and properties (see Section 2.1.1) of various models for specific learning tasks. Popular examples include *RobustBench* [Cro+21] (for adversarial training with images), the *Monash Time Series Forecasting Archive* [God+21], the *LLM-Perf* [MP23] and *Beyond the Imitation Game* [Sri+23] benchmarks (for NLP), as well as *WinoGrande* [Sak+21] and *ARC*[6] [Cho19; Cho+25] (for reasoning tasks). Most benchmarks offer Application Programming Interfaces (APIs) to submit, evaluate, and score models for display on the leaderboards and usually link back to the original papers. Taking this idea a step further, the **sixth** and final reporting type comprises cross-domain evaluation *platforms*, such

[6] This challenge, in which we actively participated [Fis+20b], was already shortly mentioned in Section 1.5.

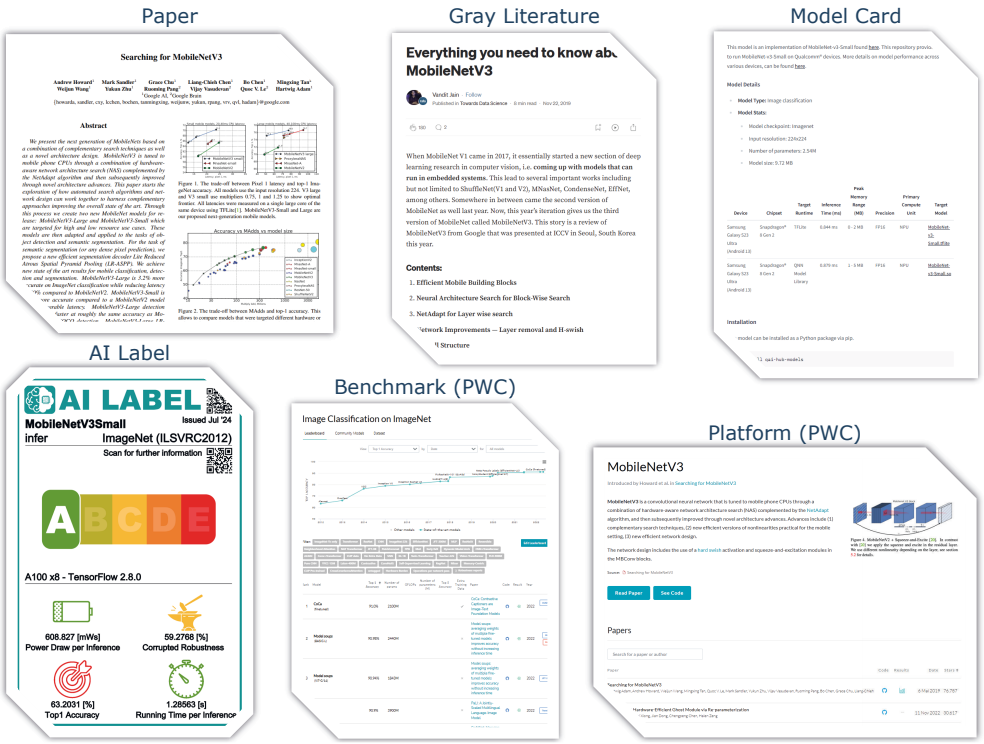

Figure 3.1: Overview for different types of reporting, here given for the `MobileNetV3` image classifier [How+19]. A similar overview was also shown to the interviewees of the later discussed labeling evaluating study [Fis+25].

as *Papers With Code* (*PWC*) [Pap24], *OpenML* [Van+14], *Hugging Face* [Hug24], or *Kaggle* [Kag24]. Instead of focusing on a single learning task, these databases offer APIs for reporting model performance results across various learning tasks and datasets, featuring many different respective leaderboards. As additional features, *PWC* links scientific papers with code repositories, *Kaggle* gained significant popularity by offering means for hosting learning task competitions (like the aforementioned *ARC* challenge), and *Hugging Face* offers useful model repository and deployment functionalities that allowed the platform to become one of the most important resources for GenAI development.

At a glance, Figure 3.1 depicts exemplary reports on the `MobileNetV3` model [How+19], representing each of the introduced types of reporting. It is important to note that reporting forms should not be considered as competing [FLM24]—today's AI developers and users come from diverse backgrounds and require different representations of information that fulfill their respective desiderata [Lan+21]. That being said, it is important to understand the issues that established reporting types suffer from, as summarized in Table 3.1. It shows the extent to which each type of reporting can be considered to be comprehensible (for non-experts), resource-aware (i.e., not only focused on predictive quality), and interactive. This analysis is based on my personal opinion and the results

Table 3.1: Drawbacks of the Established Types of Reporting

Reporting type	Comprehensible	Resource-Aware	Interactive
Papers	✗	✗	✗
Gray literature	○	○	✗
Model cards	✗	○	✗
AI labels	✓	✓	✗
Benchmarks	✗	○	○
Platforms	○	○	✓

from our AI labeling evaluation study, in which interviewees were presented with a similar overview [Fis+25]—the respective work will be discussed in Section 4.2. In most cases, a strict yes (✓) or no (✗) answer cannot be given: Gray literature and platforms for example have diverse target audiences, so to some extend they might be comprehensible for non-experts, however not in all cases. Regarding resource-awareness, Section 2.3.2 already established that the research field and resulting reports are heavily biased toward overly focusing on predictive capabilities while neglecting negative performance aspects and resource trade-offs [Bir+22; VLW25]—empirical evidence will be discussed in Section 3.4.3. In addition, a meta study found many ML papers to not sufficiently justify the investigated performance metrics [Küh+20], and model cards were also analyzed to under-report on "environmental impact, limitations, and evaluation" [Lia+24]. AI labels are the only reporting type where resource efficiency was specifically part of the design process, while conceptually not requiring any fundamental knowledge of ML and AI [Fis+22].

Interactivity can benefit user understanding by making systems more transparent, which was for example shown in the context of XAI [Bec+23]. The importance of customizability was also among the central analysis points in our labeling evaluation (see Section 4.2.3). In current reporting, interactivity (if present at all) is usually limited to sorting leaderboards and providing links to original publications. What users really need, however, are means for tailoring reports to their individual needs and use cases [Pio+21], for example by directly controlling the importance of the various reported properties. While not explicitly listed in Table 3.1, established reporting also suffers from usability issues, as many reports need to be compiled by hand, thus leaving room for subjectivity and bias (even though *Hugging Face* partially supports automated model card creation and GenAI can alleviate some of the effort). Benchmarks and open platforms are somewhat more automatized, however there is often no mechanism or control regarding how new data is entered, resulting in a lack of consistency and standardization. A recent analysis further supports the analyzed issues in reporting and concludes that AI documentation needs to become more holistic (in also informing on operational context), engaging (i.e., interactive), and automated (with less manual efforts) [Arn+24].

Note that in the original assessment [FLM24], the reporting types were differentiated based on whether they inform on singular models, specific learning tasks (i.e., benchmarks), or across different tasks (i.e., open databases and platforms). While model cards and AI

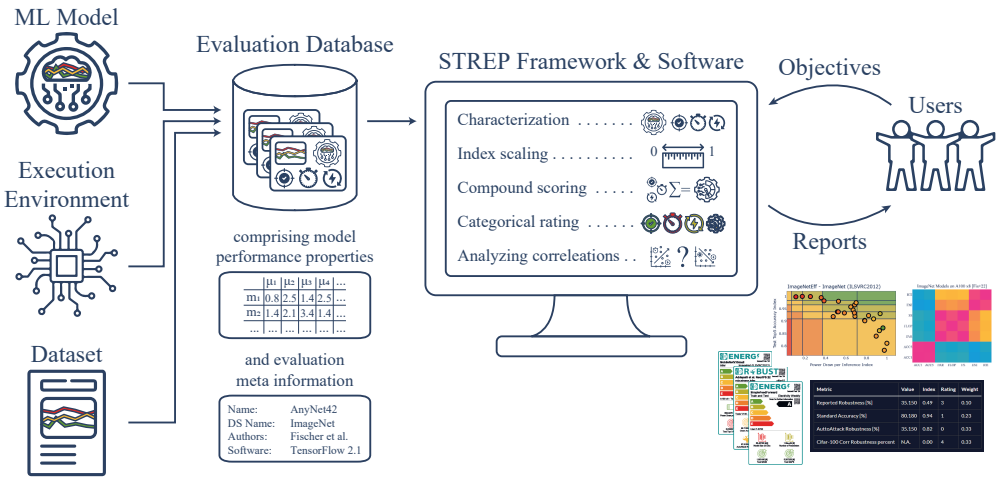

Figure 3.2: Framework for sustainable and trustworthy reporting, adapted from the visualization from the original paper [FLM24] but aligned with the contents of this thesis.

labels clearly report on individual models, this separation is somewhat more tricky to define for the other types: Papers and gray literature can discuss results on any of those levels and the named platforms do feature leaderboards as well as information on single models (sometimes even with different representations). Finally, note that the originally listed aspects of usability and reproducibility [Hut18], while obviously important for trustworthiness [Avi+21] and reporting [Hut18], were here neglected because they are not really advanced or evaluated in the following.

3.2 Methods for Better Reporting

Overcoming the aforementioned issues of reporting and addressing the pressing need for sustainability and trustworthiness requires a unified methodological framework, which is schematically displayed in Figure 3.2. Strongly inspired by the original paper [FLM24], it comprises five key steps which will be introduced in the following. A proper *characterization* of model evaluations acts as the fundament (Section 3.2.1), *index scaling* makes the performance of models comparable across multiple properties and execution environments (Section 3.2.2), *compound rating* allows to customize and interact with reports (Section 3.2.3), *categorical rating* makes results more comprehensible for informing diverse target audiences (Section 3.2.4), and *analyzing correlations* allows to identify reporting biases (Section 3.2.5)[7]. Figure 3.2 guides the following formalization and discussion of these steps.

[7] In the original STREP paper, correlation as a measure for reporting biases was only used in the experiments [FLM24], whereas it is here explicitly formalized as the fifth step.

3.2.1 Characterizing Models and Properties

The characterization of ML experiments and resulting AI models is of central importance for collecting well-structured databases of model performance and reporting these results. Any report (independent of its type) should therefore explicitly formalize which learning task T, dataset D_T, model m, execution environment E, and properties $\boldsymbol{\mu}$ it describes. The background on ML (cf. Section 2.1) already introduced formalizations for learning tasks (Definition 2.1) and model properties (Definition 2.5), which are now extended for reporting on results from an AI model *evaluation database*:

Definition 3.1. *Let $\boldsymbol{\mu} = (\mu_i)_{i \in \mathbb{P}_T}$ be the property functions for evaluating the performance of models $\{m_k\}_{k=1}^{K} \subseteq \mathcal{M}_T$ across various configurations $\{C_l\}_{l=1}^{L}$ (i.e., datasets $\{D_T\} \subseteq \mathcal{X}_T \times \mathcal{Y}_T$ and environments $\{E\} \subseteq \mathcal{E}$). Any set of evaluation results is then defined as an* evaluation database $\mathfrak{D} = \{(m_k, C_l, \boldsymbol{\mu}(m_k, C_l))\}$*, from which results can be reported in different ways.*

Connecting back to the `ImageNet` examples of Section 2.1, a tuple $(m_k, C_l, \boldsymbol{\mu}(m_k, C_l))$ would thus represent a model evaluation result that comprises various performance properties ($\boldsymbol{\mu}$) such as ACC1, ENT, or RTI for an image classifier m_k evaluated on some hardware and software (configuration C_l). Considering the aforementioned database and benchmark examples, there unfortunately is no unified naming convention, however some connections can be drawn. *PWC*, for example, also consists of evaluations for specific tasks and datasets, which however are referred to as *metrics* of evaluated *methodologies* (instead of model properties) [Pap24]. In *OpenML* terminology, these metrics are instead called *measures* from individual *runs* of specified ML *flows*, which themselves can be characterized in more detail—users can for example specify a *setup* to inform on the exact hyperparameters [Van+14]. Both report databases offer APIs for contributing results from ML experiments based on their respective terminology, however more incentives to report in sustainable ways would be desirable [FLM24]. In favor of resource-awareness, contributions should always explicitly describe model performance with regard to both resource consumption and predictive quality, as well as explicitly inform on the utilized execution environment. For comprehensibility and trustworthiness, reported results and measurable properties should be accompanied by extensive meta information, such as textual, comprehensible descriptions as well as links to further information and associated reports (e.g., corresponding papers). With the diversity of properties and execution environments, even if characterized well, it however becomes evident that comparability is non-trivial, which leads into the second step of `STREP`.

3.2.2 Archiving Comparability via Index Scaling

As mentioned in Section 2.1.4, the properties describing resource-allied performance are likely to scale with multiple levels of magnitude when performing experiments across different hardware environments. Some models like `MobileNetV2` [San+18] were optimized

for usage on edge devices instead of standard CPU or GPU machines, and generally, the running time and energy consumption might change dramatically with testing different hardware setups. While this problem is especially pronounced with regard to resource consumption, one should keep in mind that the environment choice might also impact the predictive quality, for example due to different implementations of ML logic. Comparing model behavior with respect to various properties and execution environments therefore necessitates unification, which can be established via *index scaling* [FLM24; Fis+24; FS24; SFB24; Fis+22]:

Definition 3.2. *Let $\mathfrak{D}$ be an evaluation database describing the performance of models $\{m_k\}_{k=1}^K$ across configurations $\{C_l\}_{l=1}^L$. This database can then be mapped onto an alternative* index-scaled *representation $\tilde{\mathfrak{D}} = I(\mathfrak{D})$, where I calculates all index-scaled representations of individual properties via the scaling function $\iota : \mathbb{R}_{>0} \to (0, 1]$:*

$$\tilde{\mathfrak{D}} = I(\mathfrak{D}) = \{(m_k, C_l, \tilde{\boldsymbol{\mu}}(m_k, C_l))\} = \{(m_k, C_l, \iota(\mu_i(m_k, C_l)))_{i \in \mathbb{P}_T}\} \tag{3.1}$$

For any specific property μ_i of a model m and configuration C, the index-scaled value is computed via:

$$\tilde{\mu}_i(m, C) = \iota(\mu_i(m, C)) = \left(\frac{\mu_i(m^*, C)}{\mu_i(m, C)}\right)^{\sigma_i}, \text{ with } m^* = \underset{m' \in \{m_k\}_{k=1}^K}{\arg\min}\ \mu_i(m', C) \cdot \sigma_i \tag{3.2}$$

The index-scaled values $\tilde{\boldsymbol{\mu}}$ are subject to the environment and dataset at hand (encoded via the configuration) and describe the practical model performance relatively; the higher the value, the closer it is to the best empiric result, which receives $\tilde{\mu}_i(m^*, C) = 1$. On the index scale, properties can be easily compared: A value of 0.8 always indicates that the method achieves 80% of the best empirically known result, regardless of which environment was used for deployment or whether this property wants to be minimized or maximized (thanks to σ_i, see Definition 2.6).

Obviously, Equation (3.2) would fail in cases where $\mu(m, C) = 0$, for example when facing a model with zero accuracy. For this reason, Definition 2.5 assures that property function values are strictly positive—any implementation of index scaling needs to handle respective outlier cases. In the publication that first introduced this scaling [Fis+22], index values were computed based on the performance of a globally defined reference model m^*. This approach could still be viable to compare performances in relation to an established baseline model. With this approach, $\tilde{\mu}(m, C) < 1$ and $\tilde{\mu}(m, C) > 1$ indicate worse and better performance than the baseline, respectively. However, choosing m^* as the empirically best-performing model for this property neatly solves the issue of choosing the reference [FLM24], and moreover allows for an easier aggregation of index-scaled values, which now become restricted to the unit interval $(0, 1]$. Naturally, when new results are added to the database $\mathfrak{D}$, it is imperative to check whether new empirically best performances were observed, upon which the index values $\tilde{\mathfrak{D}}$ need to be updated.

The introduced concept of index scaling allows to report and compare the performance of models in relation to different properties and environments [FLM24; Fis+22]. In our work on stress-testing edge accelerators, we use the same approach to define and discuss scaled and unscaled units under test [SFB24] (these results will be discussed in Section 3.4.2). Index scaling is also of central importance for performing sustainable meta-learning, which will be discussed in Chapter 5. Note that index values are not superior than the original measurements—many users might find the real values more intuitive, but the index scale offers an alternative representation that is more suitable for comparing model performance across properties, datasets, and environments. In the favor of interactivity, users can thus be empowered to select the value scale that is best suited for their investigation, either exploring real values, index-scaled values, or index scaling with respect to a reference baseline. This leads into the third step toward STREP, which focuses on reducing the complexity of model performance results based on user preferences.

3.2.3 Interactive Reporting via Compound Scoring

AI can be helpful for diverse applications and use cases, hence today's practitioners need to be able to infuse reporting frameworks with their own priorities and requirements in interactive ways. To give some examples, consider safety-critical applications, where highly robust models are needed [Cro+21]; real-time use cases, which specifically demand fast inference [Bus+18]; or edge deployment scenarios, whose tight memory constraints require models with low memory footprints [SFB24]. While many established reporting frameworks allow for sorting results based on a single property, it would be desirable to score model performance in a more sophisticated way. With index-scaled properties, this can be achieved via interactive *compound scoring*:

Definition 3.3. *Let $\tilde{\boldsymbol{\mu}}$ be the index-scaled properties of some model m evaluated for some configuration C. The model performance can then be assessed with respect to a user-defined objective $\Omega = (\omega_i)_{i \in \mathbb{P}_T}$ (i.e., weights for each property) via a weighted* compound score *$S_\Omega(m, C)$ that is defined as:*

$$S_\Omega(m, C) = \sum_{i \in \mathbb{P}_T} \omega_i \cdot \tilde{\mu}_i(m, C), \text{ with } \sum_{i \in \mathbb{P}_T} \omega_i = 1 \text{ and } 0 \leq \omega_i \leq 1 \forall \omega_i \tag{3.3}$$

The normalization of weights ω_i in combination with the unified value scale of properties $\tilde{\mu}_i$ ensure that the compound score is also bounded to the unit interval, i.e., $0 < S_\Omega(m, C) \leq 1$. A case of $S_\Omega(m, C) = 1$ would indicate that the model m is superior to all other empiric results in every single aspect, which in practice—and recalling the NFL theorem [WM97]—is highly unlikely. By adjusting the weights of the objective Ω, users can interactively control the importance of every single performance aspect with respect to their use case preferences, such as low error rates, fast running time, or high robustness. Note that with property groups (as presented in Table 2.1) and appropriately normalized weights ω_i, one can also easily compute *partial scores* for individual groups of properties, like all quality-allied properties:

Definition 3.4. *In extension to Definition 3.3, let $\{\mu_i\}_{i\in\mathbb{P}_Q}$ be the group (or subset) of properties (i.e., $\mathbb{P}_Q \subseteq \mathbb{P}_T$) that describe the predictive quality of a model m for some configuration C. The respective* quality score $S_{Q,\Omega}(m, C)$ *can then be computed via:*

$$S_{Q,\Omega}(m, C) = \sum_{i\in\mathbb{P}_Q} \omega_i^Q \cdot \tilde{\mu}_i(m, C), \text{ with the group weights } \omega_i^Q = \frac{\omega_i}{\sum_{j\in\mathbb{P}_Q} \omega_j} \tag{3.4}$$

Other partial scores can be defined accordingly, such as the resource score $S_{R,\Omega}(m, C)$*, which is based on all properties $\{\mu_i\}_{i\in\mathbb{P}_R}$ that describe aspects of resource consumption.*

As before, normalization ensures that partial scores such as $S_{Q,\Omega}$ and $S_{R,\Omega}$ are bounded to the unit interval. Regardless of the specific learning task and properties at hand, these scores allow to investigate important trade-offs in model performance, for example in terms of predictive quality versus resource consumption. While this trade-off is centrally investigated in this thesis, Definition 3.4 could also be used to aggregate other groups of properties (e.g., complexity, robustness, or fairness metrics).

Several other opportunities exist for interactive reporting that go beyond compound scoring. For example, reporting frameworks could present results via intuitive and responsive user interfaces. In addition to static text and tables, they can incorporate interactive plots [Plo24] and perhaps even methods of visual analytics [Cui19] to display data and performance results in intuitive ways. In this context, both the index scale as well as the original measurement scale of properties can be helpful for practitioners. Reporting frameworks can depict results side-by-side or allow users to switch between them, allowing them to understand the transformation and investigate model performance absolutely and relatively. Lastly, the insights from Table 3.1 can be used to showcase results at different levels of understanding, which also leads into the fourth step of STREP.

3.2.4 Establishing Comprehensibility via Categorical Rating

As explained in Section 3.1, the different types of reporting address different target audiences and AI stakeholders. To give some examples, in-depth scientific papers as well as library documentations are mostly helpful for ML engineers and experts, less knowledgeable practitioners might be interested in more practically oriented blogs or model cards, and users of AIaaS only need very abstract summaries that describe the most important model properties in comprehensible ways—this will be further evidenced in Section 4.2, based on interviews with different AI practitioners [Fis+25]. As a result, the next step toward better reporting therefore requires to explicitly understand the demands of different audiences and represent the information at their respective level of understanding.

However, one central problem in addressing non-experts is that evaluation databases requires at least some fundamental knowledge and intuition of math and numbers, if not even a certain amount of ML expertise (as for example introduced in Section 2.1.4). Index scaling brings some alleviation due to the unified unit scale, however an intuitive interpretation or comparison of results might still be difficult for non-technical users. In

analogy to established high-level reporting systems, index scaling however also neatly allows to discretize the properties and communicate them as *categorical ratings*:

Definition 3.5. *Let $\{\tilde{\mu}(m_k, C)\}_{k=1}^{K}$ be the K index-scaled observations from a property function μ across a set of models $\{m_k\}_{k=1}^{K}$ evaluated for a fixed configuration C. Then a respective empirical cumulative distribution function can be defined as:*

$$F_{\tilde{\mu}}(\tilde{\mu}(m', C)) = \frac{1}{K} \sum_{k=1}^{K} \mathbf{1}(\tilde{\mu}(m_k, C) \leq \tilde{\mu}(m', C)), \tag{3.5}$$

where $\mathbf{1}(\cdot)$ is the indicator function. Each value $\tilde{\mu}(m', C)$ can then be mapped to a categorical rating $\check{\mu}(m', C)$ *based on user-controllable quantile* thresholds $\boldsymbol{\tau} = \{\tau_1, \tau_2, \tau_3, \tau_4\}$*, using a mapping function $\xi : (0, 1] \rightarrow \{\textbf{\textit{A}}, \textbf{\textit{B}}, \textbf{\textit{C}}, \textbf{\textit{D}}, \textbf{\textit{E}}\}$:*

$$\check{\mu}(m', C) = \xi(\tilde{\mu}(m', C)) = \begin{cases} \textbf{\textit{A}}, & \text{if } F_{\tilde{\mu}}(\tilde{\mu}(m', C)) > \tau_1 \\ \textbf{\textit{B}}, & \text{if } \tau_2 < F_{\tilde{\mu}}(\tilde{\mu}(m', C)) \leq \tau_1 \\ \textbf{\textit{C}}, & \text{if } \tau_3 < F_{\tilde{\mu}}(\tilde{\mu}(m', C)) \leq \tau_2 \\ \textbf{\textit{D}}, & \text{if } \tau_4 < F_{\tilde{\mu}}(\tilde{\mu}(m', C)) \leq \tau_3 \\ \textbf{\textit{E}}, & \text{if } F_{\tilde{\mu}}(\tilde{\mu}(m', C)) \leq \tau_4 \end{cases} \tag{3.6}$$

In similar fashion, ratings can be defined and calculated for the compound and group scores, which are correspondingly projected onto $\check{S}_{\Omega}(m, C)$, $\check{S}_{Q,\Omega}(m, C)$, and $\check{S}_{R,\Omega}(m, C)$. A complete index-scaled evaluation database can thus be also transformed into a rated representation, denoted as $\check{\mathfrak{D}} = \Xi(\tilde{\mathfrak{D}})$

In short, the representation obtained from applying Ξ allows to communicate the relative performance of a model, either in terms of single properties, groups of properties, or for the complete model, via more intuitive color-coded ratings. The number of categories as well as the respective colors are here chosen in accordance with the EU Nutri-scoring system [Org22], however could also be easily changed. Equally sized quantiles are suggested as the default setting (i.e., $\boldsymbol{\tau} = \{0.8, 0.6, 0.4, 0.2\}$), however could also be adapted to other preferences. Moreover, the meaning of properties and groups can be abstracted via icons, i.e., a stopwatch for running time, a battery for energy draw, or a target circle for accuracy, which can then be color-coded to denote the respective performance of the model at hand.

In the context of addressing non-technical target audiences, this methodology is key for representing model performance results via abstract AI labels [Fis+22], as already mentioned in Table 3.1. Chapter 4 will explore the concepts for AI labeling in more detail [Mor+22] and also summarize findings from our qualitative evaluation [Fis+25]. It is important to note that labeling might not satisfy all needs—certain practitioners require more technical details, so good reporting requires to link and support multiple reporting types. This is another aspect allowing for interactivity, such that users of a reporting framework might start off at the most basic level but can work their way into more technical levels.

3.2.5 Detecting Reporting Biases via Analyzing Correlations

As the last step for STREP, we acknowledge that reporting and the respective evaluation database should ideally be free of biases. They could for example occur when certain aspects of model performance are neglected—following the NFL theorem [WM97], one can safely assume that different AI models will have pros and cons, which should be transparently reported. To give an example, models could sacrifice fast inference (RTI) or small number of parameters (MP) for higher predictive accuracy (ACC1), but such trades need to be explicitly discussed. To investigate biases, index scaling not only allows for comparing models along these dimensions, but can also be used to mathematically assess the *correlation* of different properties:

Definition 3.6. *Let $\tilde{\mu}_i(m_k, C)$ and $\tilde{\mu}_j(m_k, C)$ be two distinct, index-scaled property functions measured across a set of models $\{m_k\}_{k=1}^K$ for a specific configuration C. Their* Pearson correlation coefficient *$r(\tilde{\mu}_i, \tilde{\mu}_j, C)$ quantifies the linear relationship between these properties and is computed as:*

$$r(\tilde{\mu}_i, \tilde{\mu}_j, C) = \frac{\sum_{k=1}^{K}(\tilde{\mu}_i(m_k, C) - \bar{\tilde{\mu}}_i(C))(\tilde{\mu}_j(m_k, C) - \bar{\tilde{\mu}}_j(C))}{\sqrt{\sum_{k=1}^{K}(\tilde{\mu}_i(m_k, C) - \bar{\tilde{\mu}}_i(C))^2 \sum_{k=1}^{K}(\tilde{\mu}_j(m_k, C) - \bar{\tilde{\mu}}_j(C))^2}} \tag{3.7}$$

In this context, $\bar{\tilde{\mu}}_i$ and $\bar{\tilde{\mu}}_j$ are the mean values of the respective properties across all models:

$$\bar{\tilde{\mu}}_i(C) = \frac{1}{K}\sum_{k=1}^{K}\tilde{\mu}_i(m_k, C) \text{ and } \bar{\tilde{\mu}}_j(C) = \frac{1}{K}\sum_{k=1}^{K}\tilde{\mu}_j(m_k, C)$$

The correlation between properties ranges from -1 (perfect inverse correlation) to 1 (perfect direct correlation), with 0 indicating no observable linear relationship. For grouped properties as given in Table 2.1, positive correlation is expected within groups (e.g., between ACC1 and ACC5), while less, no, or even negative correlation (i.e., trades) should occur between properties of different groups (e.g., RTI and ACC1). Index scaling ensures that all properties want to be maximized for improvement, otherwise Equation (3.7) could not be applied so easily. Moving beyond two properties, one can also compute and investigate the *correlation matrix* $\mathfrak{R}$ for a complete database:

Definition 3.7. *Let $\tilde{\mathfrak{D}} = \{(m_k, C_l, \tilde{\boldsymbol{\mu}}(m_k, C_l))\}$ be an index-scaled evaluation database of model performance properties. The* correlation matrix *$\mathfrak{R}$ then constitutes the pairwise correlations between any two properties μ_i and μ_j across all evaluated models and configurations, i.e.,*

$$\mathfrak{R} = [\bar{r}_{i,j}], \text{ with } \bar{r}_{i,j} = \bar{r}(\tilde{\mu}_i, \tilde{\mu}_j) = \frac{1}{L}\sum_{l=1}^{L} r(\tilde{\mu}_i, \tilde{\mu}_j, C_l) \tag{3.8}$$

This correctly takes into account that model properties and the resulting correlations might be different across investigated environments and datasets (even though the correlation is

expected to be only marginally affected by the environment choice). From the pairwise computation of $\bar{r}_{i,j}$, it directly follows that $\mathfrak{R}$ is symmetric, with diagonal values of 1 (due to self-correlation) and all entries being bounded to the range $[-1, 1]$. Investigating the off-diagonal entries of $\mathfrak{R}$ neatly allows to finally assess the *level of bias* in the evaluation database:

Definition 3.8. *Let $\mathfrak{R} = [\bar{r}_{i,j}]$ be the correlation matrix of an index-scaled evaluation database $\bar{\mathfrak{D}}$. The* level of bias *in this database can then be assessed via the mean and standard deviation of correlation, i.e., $\bar{\mathfrak{R}}$ and std$(\mathfrak{R})$, which are calculated over the off-diagonal entries in R:*

$$\bar{\mathfrak{R}} = \frac{1}{\binom{I}{2}} \sum_{1 \leq i < j \leq I} [\mathfrak{R}]_{i,j} \text{ and } std(\mathfrak{R}) = \sqrt{\frac{1}{\binom{I}{2}} \sum_{1 \leq i < j \leq I} ([\mathfrak{R}]_{i,j} - \bar{\mathfrak{R}})^2} \tag{3.9}$$

From the bounded correlation coefficients in $\mathfrak{R}$, it naturally follows that the mean correlation $\bar{\mathfrak{R}}$ is also bounded to $[-1, 1]$, while the standard deviation std$(\mathfrak{R})$ will always be in the interval $[0, 1]$. Very high values for $\bar{\mathfrak{R}}$ will be encountered for strongly biased databases, which only inform on performance properties with high correlation and do not feature any trade-offs. In contrast, lower values of $\bar{R}$ indicate a more diverse set of investigated properties that also entail weak or anti-correlation, which as explained earlier, is expected for sustainable reporting. The standard deviation std$(\mathfrak{R})$ helps to further interpret the bias, with a high value indicating that both positive and negative correlation is found within $\mathfrak{D}$. This concludes the STREP methodology, allowing practitioners to compare model performance results, communicate them to non-experts, and investigate correlations and biases.

3.3 Reporting Software Implementation

The introduced methodology can either be used for refining existing reporting frameworks or for constructing new ones. As an integration into existing systems is not trivial, the original STREP paper was instead accompanied by a framework implementation [FLM24]. Originally designed as an *AI Energy Label Exploration tool* (ELEx) [Fis+22], the STREP software has matured over the years. Today, it allows users to explore different evaluation databases and automatically create reports for investigating and communicating the performance and trade-offs of various AI models.

The following will present the details of the STREP software, which can be found at `https://github.com/raphischer/strep`. Note that the software is both a proof of concept and work in progress, and as such, is subject to change. In the future, the information offered by STREP may diverge from the results in this thesis and the original papers, however the current repository state with respect to the following investigations will be frozen to a permanent branch. In addition to the reporting framework, the linked repository also entails scripts and data to re-generate all figures used in this thesis.

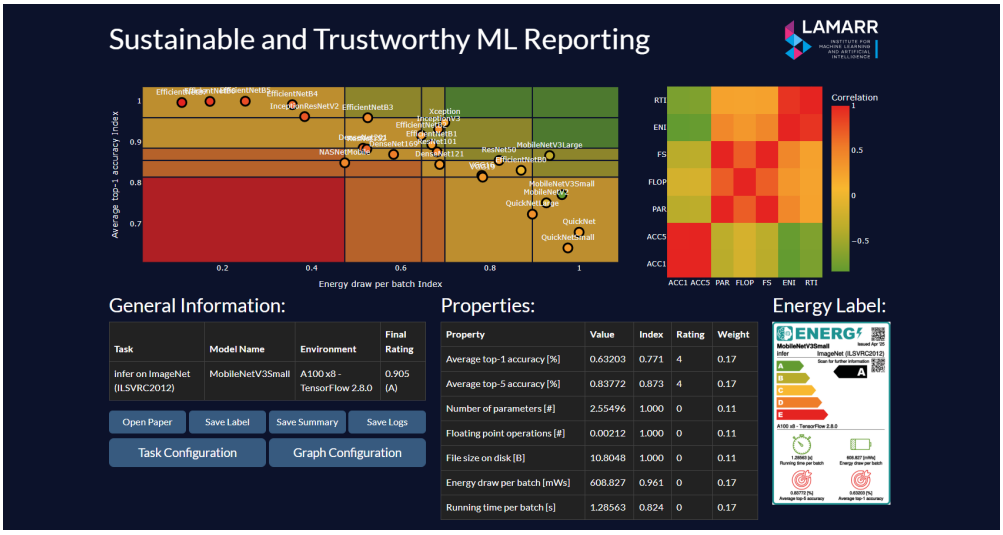

Figure 3.3: User interface of the implemented STREP reporting framework.

3.3.1 Core Features

STREP is implemented as an open-source Python 3.10 software library that comes with a frontend exploration dashboard, which can either be accessed on a public webpage or launched locally. This interactive tool invites users to explore either pre-compiled or custom assembled databases of empirical model performance. In addition, developers can easily import the methodological functionality of STREP and use it for their own implementations. The library is implemented in a completely task-agnostic way, so users can utilize it to investigate any arbitrary collection of ML or AI evaluation results, as long as it follows the simple table layout that STREP expects. It is closely aligned with the characterization introduced in Section 3.2.1 and exemplarily displayed in Table 3.2, through the example of ImageNet model performance results [Fis+22].

The interactive dashboard is depicted in Figure 3.3 and most prominently features a two-dimensional scatter plot. It can be used to display trade-offs between two properties (or property groups) of the currently selected database, with each scatter point representing the performance of a single model. On the right, a heatmap plot visualizes the correlation matrix for the currently selected learning task and dataset (introduced in Definition 3.7). Upon hovering over any model performance point, the lower half of the dashboard is updated to display general information and all properties in the form of tables, as well as a corresponding AI label, which is generated on the fly (details will be explained in Section 4.1).

The selected database, learning task, dataset, and execution environment can be changed in the task configuration menu. Per default, the plot displays the most important quality and resource properties (based on given group information and property weights). Users can use the graph configuration menu to change the displayed properties, weights, and

Table 3.2: Table Layout for STREP Evaluation Databases Like *ImageNetEff22*

task (T)	dataset (D_T)	environment (E)	model (m)	μ_{ACC1}	μ_{ENI}	μ_{RTI}	···
infer	ImageNet	A100 x8	EfficientNetB2	0.75	0.91	1.63	···
infer	ImageNet	A100 x8	VGG16	0.67	0.75	1.17	···
infer	ImageNet	A100 x8	MobileNetV3Small	0.63	0.61	1.29	···
infer	ImageNet	A100 x8	ResNet50	0.70	0.71	1.36	···
infer	ImageNet	RTX 5000	EfficientNetB2	0.75	0.49	2.25	···
infer	ImageNet	RTX 5000	VGG16	0.67	0.52	1.91	···
infer	ImageNet	RTX 5000	MobileNetV3Small	0.41	0.14	0.68	···
infer	ImageNet	RTX 5000	ResNet50	0.70	0.44	1.65	···
...	...	...	...	...	...	...	...

scale (index or real-valued, see Section 3.2.2). Via designated buttons, the users can also download the currently displayed data or access the original paper (if provided in the database meta information).

3.3.2 Internal Details

Internally, the software primarily leverages `pandas` for handling evaluation databases [tea20; McK10] and `Plotly` and `Dash` for creating and managing the interactive dashboard and plots [Plo24]. Native `Python` dictionaries are used to store and link additional information, such as the relations between properties and learning tasks. Index scaling, compound scoring, categorical rating, and correlation detection, as introduced in Section 3.2, are efficiently implemented with `NumPy` logic [Har+20] and explicitly handle outlier cases such as negative property values. The public webpage is currently offered by `Render` [Ren25] and automatically updates with any changes in the repository.

The current repository structure of STREP is visualized in Figure 3.4. On top level, it contains the `main` script for starting the software application, the library folder (***strep***), two directories for the databases and additional materials, and some auxiliary files. The materials folder contains additional data, figures from papers and this thesis, as well as scripts to automatically generate them. The whole functionality for running the STREP methods introduced in Section 3.2 is implemented in the `index_scale` script. It is extremely lightweight and thus can be easily used in any context, because it only depends on `NumPy`, `pandas`, and some utility functions from the respective script. The evaluation databases are represented as large `pandas` dataframes, for which intricate grouping and indexing routines are implemented to access and handle the evaluations tables. They comprise model performance results for specific tasks, datasets, and environments (i.e., combinations of $T \times D_T \times E$), as stored in respective table columns displayed in Table 3.2.

Currently, the STREP repository offers seven pre-assembled databases, which are described in Table 3.3. The first two databases contain information on the resource and quality efficiency of pre-trained `ImageNet` models [Fis+22], with the latter specifically evaluating the inference efficiency on USB accelerator hardware [SFB24]—these databases will be

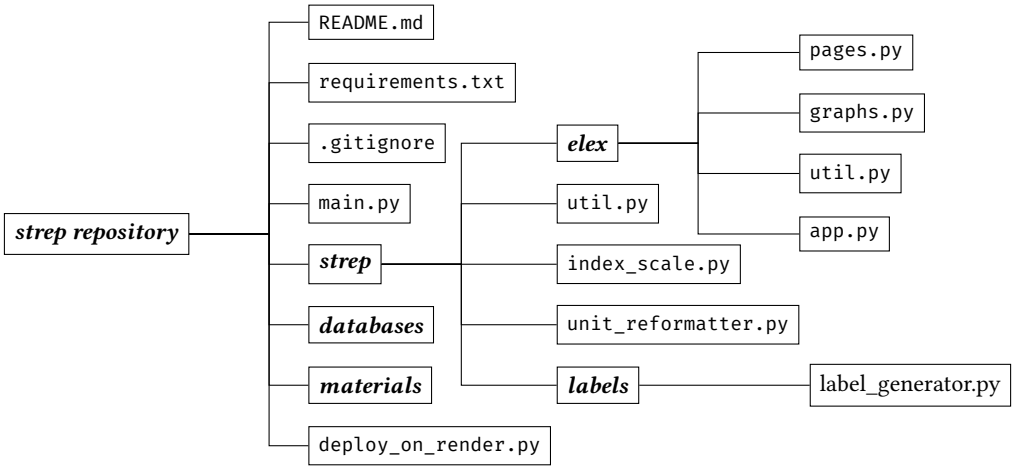

Figure 3.4: Current repository structure of STREP. The main script allows to load different databases, either provided by the user or the ones from the ***databases*** directory. The evaluation data is processed with the index_scale library and visualized via the ***elex*** application.

explored in Section 3.4.1 and Section 3.4.2, respectively. *XPCR* and *MetaQuRe* were assembled for the later discussed works on sustainable AutoML and meta-learning [FS24; Fis+24]. In these experiments, the performance of a fixed set of models was evaluated over a wide range of datasets, which Section 5.2 and Section 5.3 will investigate in more depth. The remaining three databases are extracted from respective online repositories via crawler scripts that can also be found in the STREP repository. Incompleteness denotes the average amount of missing property values for each $T \times D_T \times E$ table: In *ImageNetEff22*, the training results for some models are missing, whereas the online databases are highly incomplete due to their internal lack of structure (e.g., differently spelled properties, only partially reported model performance results, ...). As the *PWC* database is extremely vast and sparsely populated, it was narrowed down to evaluations with at least two numeric properties and ten investigated models. The resulting database is named *PWC_FULL*, whereas *PWC* only features the 20 evaluations with the highest amount of performance results. Note also that the evaluation tables in *PWC* were assembled from various papers that performed their experiments on individual execution environments, however as the platform offers no unified way to report on these specifics, STREP handles them as unknown (N/A). *RobBench* was extracted from the respective benchmark on training robust DNNs via adversarial examples [Cro+21].

Additional meta information about each database (e.g., names, descriptions, abbreviations, links to further information) is provided via JSON files and automatically included for the available visualizations. In addition, the meta information file for reported properties can be used to weight the properties and assign them to groups, with the software also supporting default assignments and automated normalization of weights. The label generator also incorporates the available meta information and produces labels with the help of

Table 3.3: Overview of Evaluation Databases Currently Offered With STREP

Database	#tasks	#ds	#envs	#models	#props	incompl.
ImageNetEff22 [Fis+22]	2	1	8	31	9	10.03%
EdgeAccUSB [SFB24]	1	2	9	26	11	0%
XPCR [FS24]	1	18	1	11	9	0%
MetaQuRe [Fis+24]	1	200	4	10	10	0%
PWC [Pap24]	15	19	N/A	3852	97	42.88%
PWC_FULL [Pap24]	360	761	N/A	17065	1276	23.49%
RobBench [Cro+21]	1	3	1	151	9	46.9%

ReportLab [Rep24] and PyMuPDF [Art24], basically creating downloadable documents by rendering texts and images onto a canvas.

All these technical details are hidden by the Dash user interface. For faster responsiveness, this application handles an internal state with the current configuration and data to display. Upon interaction events captured by Dash callbacks, the respective data is accessed, updated (if necessary), aggregated, and visualized.

3.4 Practical Investigations

Having introduced methods and implementation details for STREP, we now explore their practicability via different experimental investigations. The first experiment shows how ImageNet model performance can be understood and reported in multi-dimensional ways, as proposed in the original paper on assessing ML energy efficiency [Fis+22]. After that, the STREP methodology is applied to performance evaluations obtained from deploying these CV models on edge accelerator hardware, for which practical trade-offs and implications are investigated [SFB24]. The section closes with a discussion of biases in reporting databases, which is inspired by some experimental results from the STREP paper [FLM24].

While the originally assembled evaluation databases are used for the following investigations (see Table 3.3), the displayed plots were generated with the current version of STREP, which can result in slight differences to the original papers. For ensuring a fair balance between predictive quality and resource consumption, the properties in all experiments of this chapter are weighted and rated as the STREP default, i.e., using equal weights $\Omega = (\omega_i)_{i \in \mathbb{P}_T}$ within each group (quality or resources) and all groups equally contributing to the compound score, as well as equally sized quantiles for categorical ratings. Readers are invited to dig deeper with the STREP software, where the publicly available exploration tool allows to interactively explore the evaluation databases and create similar figures without requiring any local code execution.

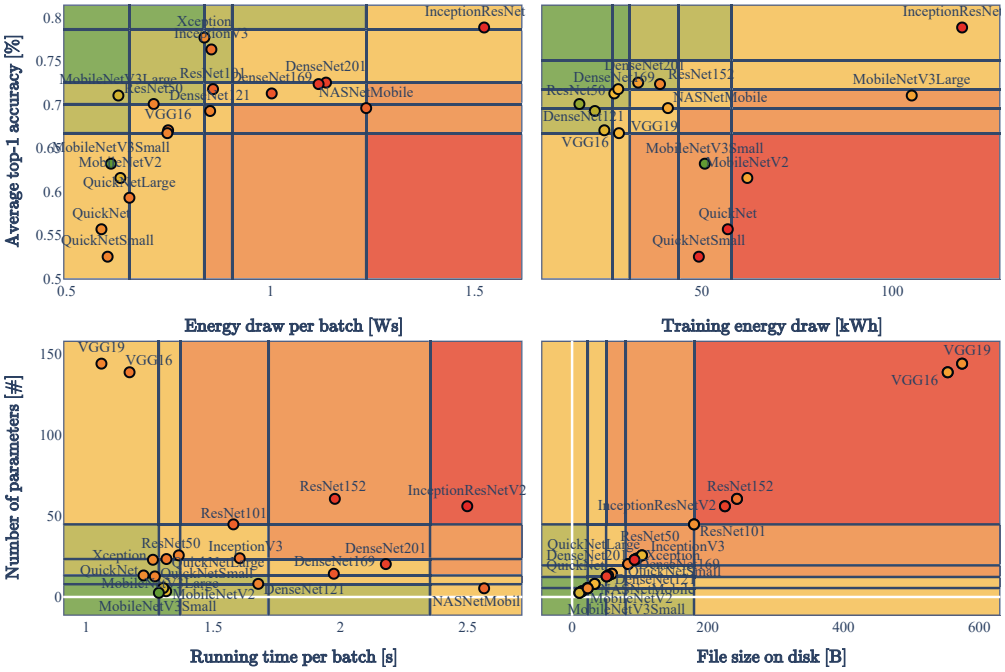

Figure 3.5: Trade-offs between accuracy and resource requirements across various `ImageNet models`, with clear property (anti-)correlations and scatter points color-coded by overall model scores.

3.4.1 Multi-Dimensional Model Performance

We start our experimental investigations with respect to the exemplary task of using DNNs to classify images from the internet. Connecting back to the examples in Chapter 2, we investigate models that were originally created by performing DL on `ImageNet` data [Den+09]. Our corresponding paper [Fis+22] assessed and analyzed the resource efficiency of 26 pre-trained `ImageNet` models, including `VGG` variants [SZ15], `ResNets` [He+16a; He+16b], `DenseNets` [Hua+17], `MobileNets` [How+17; San+18; How+19], `EfficientNets` [TL19], and others [Cho17; Gho17; Zop+18]. The *ImageNetEff22* database was assembled by testing their performance across multiple execution environments, as already listed in Table 3.3. The full code for re-assembling the evaluation database by performing the individual experiments can be found at `https://github.com/raphischer/imagenet-energy-efficiency`. The resulting multi-dimensional model performance landscape will now be discussed by leveraging the `STREP` methods introduced in Section 3.2.

To start off, Figure 3.5 displays the trades happening between between different performance properties as introduced in Table 2.1, with each scatter point representing a pre-trained `ImageNet` model as offered by `Keras` with the `TensorFlow` backend [Aba+16]. Note that these plots do not list models that could not be evaluated for resource consumption during training. When for example comparing energy (ENI) versus accuracy (ACC1) (upper left), a Pareto front can be observed, which showcases how improving

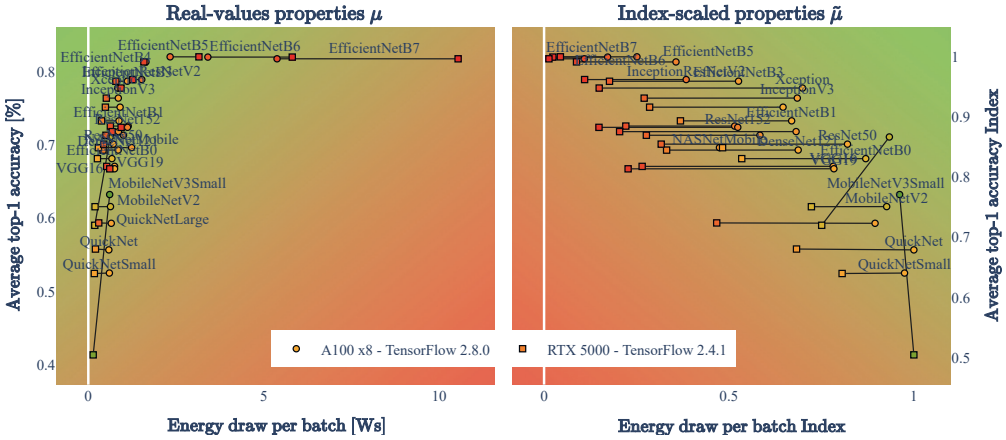

Figure 3.6: Runtime versus accuracy comparison for models across two execution environments, connected by lines. The right plot highlights how index scaling simplifies the analysis of relative performance and unifies environment differences for both properties.

quality (y-axis) requires to also invest more resources during inference (x-axis), linking back to Section 2.1.4 and Figure 1.2. At same time, we see in the upper right that the energy required for training these models (ENT, x) in comparison with quality reveals less structured patterns—the most accurate model was also the most expensive to train, but some other models perform well despite their extremely low training energy consumption. The lower left displays how most models seem to be slightly correlated in terms of their running time per processed batch (RTI, x) and respective number of model parameters (MP, y). Correlation between these measures is to be expected however not guaranteed, as for example evidenced by the `VGG` models. An even stronger correlation can be observed when comparing the number of parameters with the model file size (FS), which also makes perfect sense. Overall, these findings showcase that the practical performance of any model is of complex and multi-dimensional nature, which needs to be reported in a transparent and resource-aware way. The model performance can be unified with `STREP` methods like index scaling, compound scoring, and categorical rating, which were here already used to determine the scatter point colors (the greener the point, the higher the overall model score) and rating quantiles (indicated by the background rectangles).

To investigate the benefits of index scaling in more detail, Figure 3.6 depicts the ENI versus ACC1 comparison across two execution environments, namely a `A100` datacenter node with eight GPUs, and a standard desktop computer with an `RTX 5000` GPU. In the previous plot, the results of the `A100` environment were displayed, whereas here the performance points of each model across both environments are displayed and connected by a line. The real-valued results (left) are hard to investigate, since the two environments result in very different but overall short running times. On the other side (right), the index-scaled values allow for a much better understanding, placing each model in a relative position to the best model on each performance dimension. For instance, it shows that the fastest model (i.e., $x = 1$) is actually not the same on the two environments—`MobileNetV3Small` [How+19]

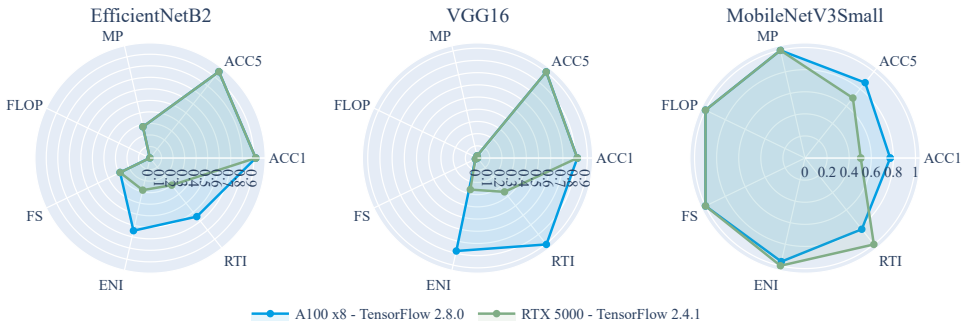

Figure 3.7: Star plots displaying the index-scaled model performance of `ImageNet` classifiers across all properties and two environments.

is fastest on the `RTX 5000`, but `QuickNet` [Gho17] is even faster on the `A100`. Moreover, the accuracy of `MobileNetV3Small` on the `RTX 5000` is only half as high (i.e., $y \approx 0.5$) compared to `EfficientNetB7` [TL19], the most accurate model at $y = 1$. This extreme loss in predictive quality could stem from differences in the hardware logic or library versions, demonstrating that sustainable reporting should transparently discuss the experimental setup. The distance between connected points allow for understanding how the environment choice impacts the relative model performance. Most models seem to be constant in their relative predictive quality, resulting in horizontal lines. In terms of energy consumption, some models are only marginally affected by the environment choice, while others are placed at very different relative points. As the boundaries of two environments will usually be different, the background here instead features a generic gradient to indicate the direction of improvement. Note also how index scaling flips the x-axis direction, as on this scale improvement is always indicated by higher values $\tilde{\mu}$.

The relative model performances across all properties can furthermore be easily visualized via star plots, as exemplarily depicted in Figure 3.7. Index scaling once again eases the unified comparison of all properties by projecting the values onto the unit scale. Higher values thus indicate better relative performance and the overall trace volume is an indicator for the compound model score (on the respective environment, or configuration). `EfficientNetB2` [TL19] (left) for example achieves very high predictive quality, however was also found to have a rather high resource consumption. Differences across the different environments can also be easily seen in this visualization—the (relative) ENI and RTI of `VGG16` [SZ15] (middle) for example are significantly impacted from the choice of environment. The resource consumption of `MobileNetV3Small` [How+19] (right) is only marginally impacted by the choice of environment, however as discussed before, it actually experiences a change in predictive quality.

Lastly, the multi-dimensional performance of `ImageNet` models can also be summarized into high level AI labels, which in analogy to the star plots are depicted in Figure 3.8. Inspired by the design of the EU energy labels, this reporting format addresses less knowledgeable users, informs on the important properties via color-coded icons obtained

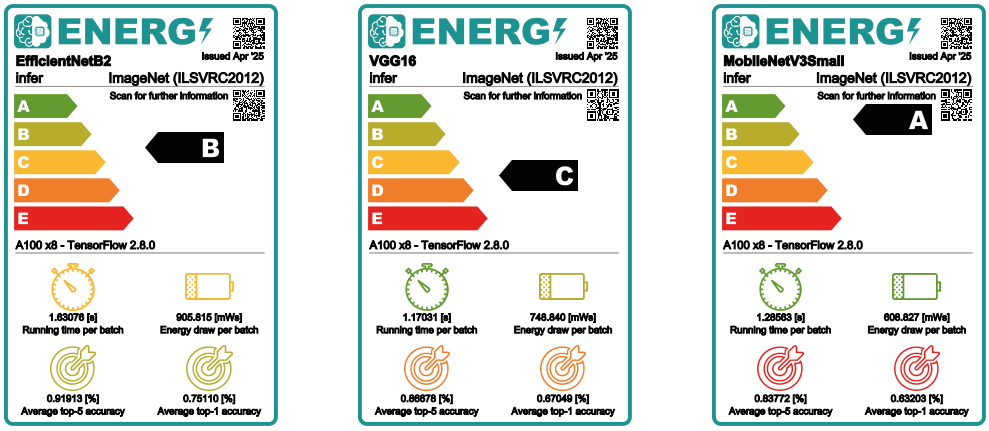

Figure 3.8: High-level AI labels generated by STREP, which inform on model properties and intricate efficiency trade-offs of ImageNet models in more comprehensible ways.

from categorical rating (Section 3.2.4), and allows to go into more technical depth via the displayed Quick Response (QR) codes. They depict the discussed trade-off of quality versus resource consumption in a more intuitive way, thus allowing non-experts to make informed decisions regarding model suitability for their use case. More details on the label concepts as well as practical insights on their feasibility for comprehensible reporting will be discussed in Chapter 4.

As a summary, this first investigation unveiled the intricate trade-offs occurring when using different DNNs for image classification. While many established reports are focused on comparing their predictive quality, the STREP methods were successfully used to explore the efficiency and quality versus resources balance of respective models. Moreover, the different representations of the experiment data allow for a more diverse exploration and even address different levels of understanding.

3.4.2 Efficiency of Edge Accelerators

The STREP methods' potential can be further demonstrated by performing a more detailed analysis of model performance evaluations across more diverse computing environments. This section therefore investigates the efficiency of using USB accelerators for deploying CV DNNs on the edge [SFB24]. The examined add-on processors are marketed as "high-speed" and "power efficient" [Cor24], allowing to upgrade standard hardware toward DL inference via simple plug-and-play USB devices. In the context of sustainability, they allow for local processing instead of cloud computing, which potentially reduces resources for data transmission and moreover has benefits for the sake of privacy. In addition, accelerators could potentially retrofit existing hardware, making local DNN deployment more affordable and eliminating the need to purchase expensive DL hardware like GPUs (which also have a high amount of embodied carbon [Wu+22]). In contrast to these promises, we however find that the few studies that actually investigated the benefits

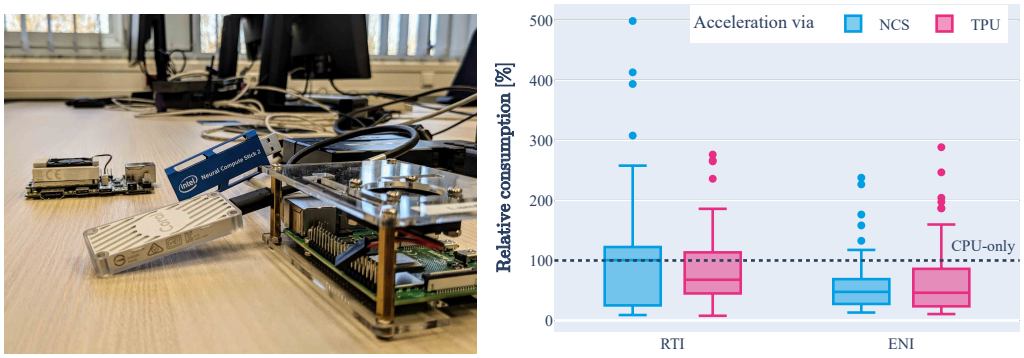

Figure 3.9: USB accelerators, as shown on the left, can potentially allow for affordable and efficient DNN inference on the edge, however in many configurations, no relative improvement can be observed for running time (RTI) and energy consumption (ENI).

of USB accelerators unfortunately only tested a very limited range of models and host systems [Reu+19; Var+21].

To address this shortcoming and properly evaluate the efficiency of accelerated inference on the edge, we conducted a novel study [SFB24] that acts as the fundament for the following investigation. In the respective experiments, the multi-dimensional performance of 37 pre-trained classification and segmentation models was explored, originally trained on `ImageNet` [Den+09] and `COCO` [JCQ23], respectively. Nine different setups were tested as execution environments: Three different host systems (a mini-tower *Desktop*, a lightweight *Laptop*, and a *RasPi* 4), which were each configured to either use the on-board CPU, the USB accelerator by `Google Coral` (TPU) [Cor24], or `Intel`'s Neural Compute Stick 2 (NCS) [Int22]. The selected host systems represent the recommended use cases of USB acceleration, as they are not designed for DL (business Laptop), slightly outdated (Desktop from 2015), or commonly used for edge computing (RasPi single-board computer).

As recommended by the manufacturers, the pre-trained `TensorFlow` [Aba+16] models were first compiled into intermediate representations and then optimized for the accelerators. Here, first concerns of usability arose, as several function mappings were unsuccessful and only half of the tested models could be successfully deployed across all processors. For comparing model performance, applicable properties from Table 2.1 were assessed, complementing the `CodeCarbon` profiling results with additional energy measurements obtained from a USB multimeter. The *EdgeAccUSB* database offered by `STREP` (Table 3.3) comprises the results of the original study, for which all implementation details can be found at `https://github.com/raphischer/edge-acc`. As an entry point to our investigations, Figure 3.9 depicts the two accelerators (left) and a high-level summary of the results (right): Accelerated edge inference in many cases indeed results in faster running times and lower energy consumption (i.e., below the CPU line), however there are also several configurations where the USB devices cannot live up to the expectations of accelerated of energy-saving inference.

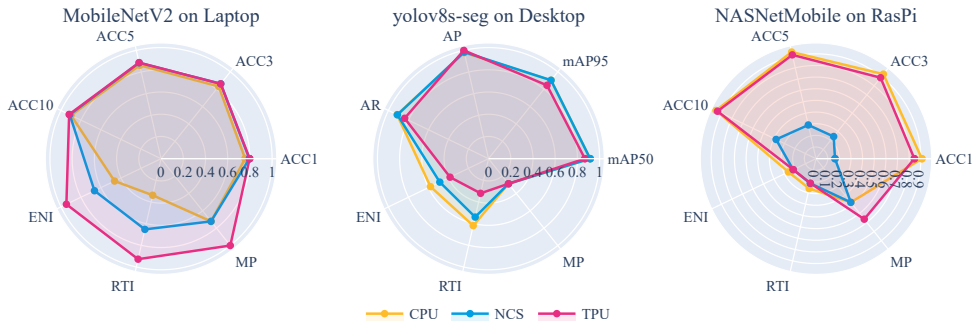

Figure 3.10: Index-scaled model performance results for deploying CV models either on the host CPU or the USB accelerators (NCS and TPU).

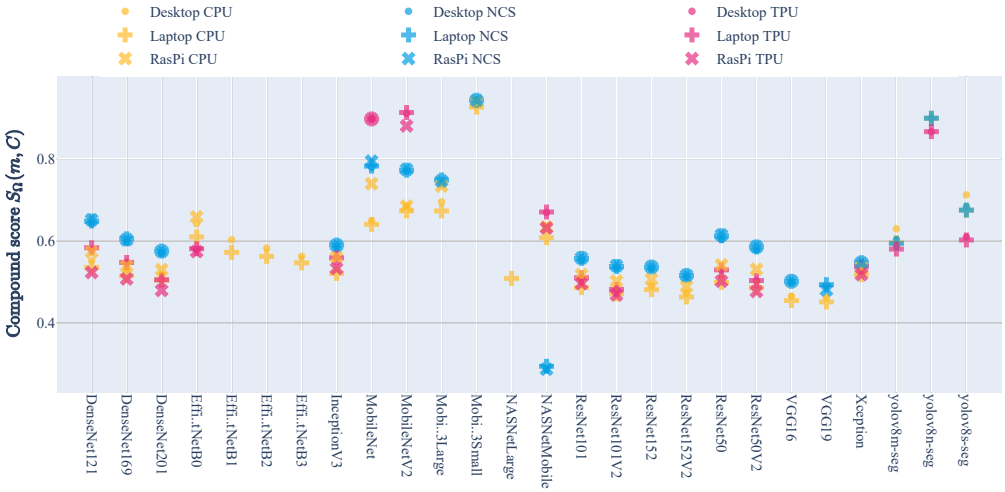

Figure 3.11: Compound model performance across all evaluated environments, showcasing several cases of rather stable behavior and also significant changes across different processors.

Digging deeper into these results, index scaling (Definition 3.2) can be utilized to compare the multi-dimensional performance of models when being deployed on the different processors, as visually shown with star plots in Figure 3.10. Remembering that better performance is indicated by higher index-scaled values, the `MobileNetV2` [San+18] performance (left) for example represents the anticipated case—RTI and ENI (relatively) improve when deployed on the USB accelerators, while the predictive quality is not affected. The `yolov8s` [JCQ23] segmentation model (middle) however performs worse on both accelerators, and for `NASNetMobile` [Zop+18] on the RasPi (right), a significantly reduced predictive quality is observed.

Via Equation (3.3), the multi-dimensional relative performance can also be aggregated into a compound score for each model and environment combination, for which Figure 3.11 gives an overview. It shows that some models, like `Xception` [Cho17], behave rather consistently across all environments, while more variance can be observed for models

such as `MobileNet` [How+17]. For many models, like for example `NASNetLarge` [Zop+18], the accelerator deployment failed and thus no respective compound score is given. In this context, one has to remember that the index values and compound scores put measured properties into relation with the best-observed results (per property). The best-scored model on the CPU and NCS is `MobileNetV3Small` [How+19], however this model could not be run on the TPU—as a result, the other TPU models receive higher compound scores.

To investigate the efficiency trade-offs in even more detail, Figure 3.12 displays the resource (x) and quality (y) scores of some models as obtained from applying Equation (3.4) to the respective property groups. Each model is represented by three connected scatter points, which indicate the performance when using one of the CPUs (different rows) or any of the two accelerators for inference. As before, we observe that the `MobileNet` variants scored well (i.e., are located on the right-hand side) because of their low resource consumption. At the same time, their relative position is strongly impacted by deploying them in different environments, because the most resource-conserving `V3` variant is not available for the TPU, resulting in higher scores for the earlier models. The more accurate models often experience a resource boost on the NCS and in some cases even achieve a higher (relative) quality, because the most accurate CPU model (`EfficientNetB3` [TL19]) could not be used with the accelerators. Index scaling clearly allows to compare how different models are impacted from the processor choice, with Figure 3.12 featuring differently shaped triangles for all models that, in some cases, however also appear similar across the three hosts.

To conclude this investigation, edge accelerators were found to provide resource benefits for certain setups, however should not be understood as magic devices that guarantee running time and efficiency improvements. From a practical standpoint, the NCS showed relatively stable performance with energy savings for most configurations, while the TPU exhibited more inconsistent behavior, sometimes outperforming the competitor device, however sometimes also compromising model accuracy. The most significant efficiency gains were observed on the RasPi; however, this host also had the most restricted model compatibility. Usability remains a central challenge, as many models could not be successfully deployed across all tested environments. Based on the efforts from our experimental preparations and analysis, our study noted that error handling and documentation of the official accelerator libraries should be improved. The results demonstrate that the `STREP` methodology is key for uncovering the relative performance trade-offs and understanding the pros and cons of USB accelerators. Contrary to marketing and expectations, our results did not find them to universally enable efficient ML inference on the edge. Exhaustive testing therefore remains essential before committing to accelerated DNN inference, especially when planning to use very large or specialized models. It should be noted that the NCS was officially discontinued [Int22], and that the TPU library has also been archived [Cor24] by now.

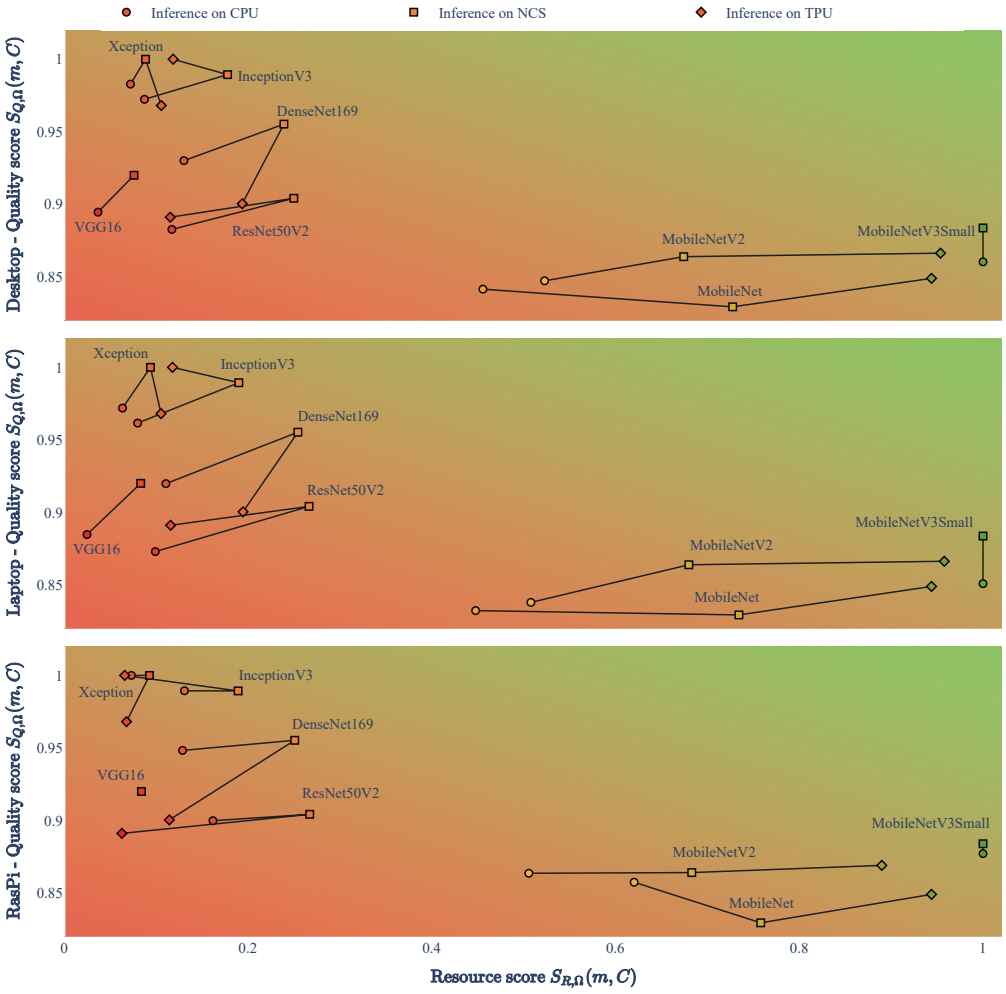

Figure 3.12: Scatter plot showing how the relative positioning of models in the resource (x) versus quality (y) group comparison are impacted from deploying the model on the host CPU or any of the two accelerators.

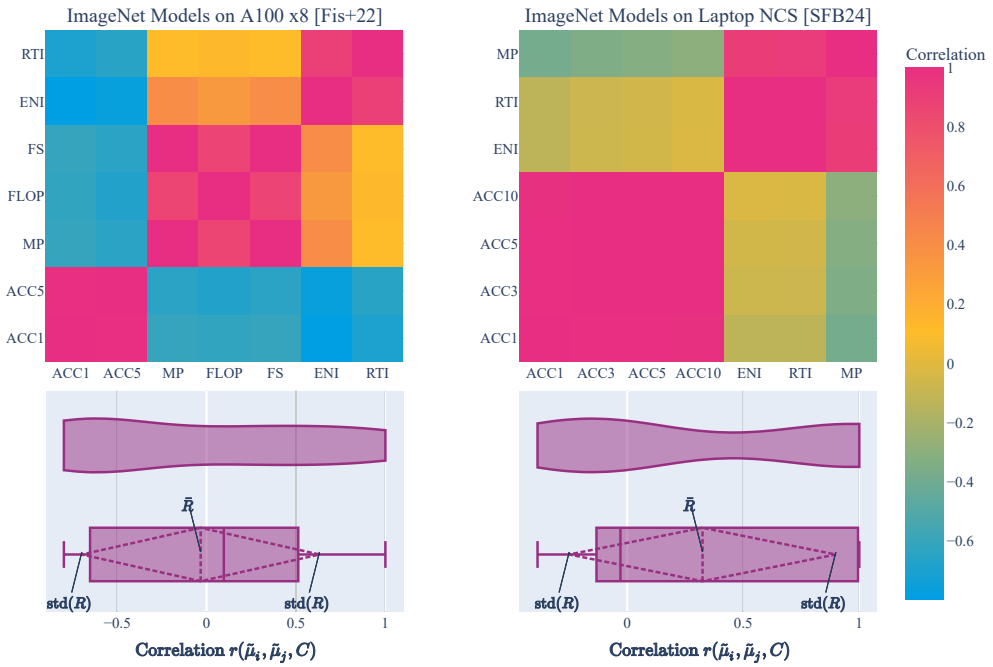

Figure 3.13: Pairwise correlations of `ImageNet` model properties observed in the previously investigated evaluation databases, showcasing an unbiased assessment of model performance (low mean correlation, high standard deviation).

3.4.3 Biases in Report Databases

Lastly, let us compare evaluation databases on a more abstract level, starting with the model performances discussed in Section 3.4.1 and Section 3.4.2. With the introduced formalizations for assessing property correlations $r(\tilde{\mu}_i, \tilde{\mu}_j, C)$ (Definition 3.6) and the respective matrices R (Definition 3.7), the contents of these databases can be summarized, as displayed in Figure 3.13. The upper half depicts the correlation matrices R, as computed from the model performance results in one specific execution environment. As explained for Equation (3.7), high values indicate strong correlation between the two respective properties, which for example can be seen for the fuchsia cells (ACC1, ACC5), (MP, FS), and (ENI, RTI). In contrast, negative correlation will be observed when two properties trade against each other, like in the blue (ENI, ACC1) cell, while green-yellow cells represent a lack of linear relationship. In the lower half, the distribution of all observed correlation values is visualized via violin and box plots (see also Definition 3.8). The latter also feature annotations for the level of bias, which is described by the mean and standard deviation across all correlation values (i.e., $\bar{R}$ and std(R)).

Analyzing this figure first of all supports the earlier assumption that trades are most likely to occur between different property groups. We see that the properties within each group are extremely correlated (near 1), while the strongest observed negative correlation is less strong (around -0.7, as shown on the color bar and lower x-axis). Interestingly,

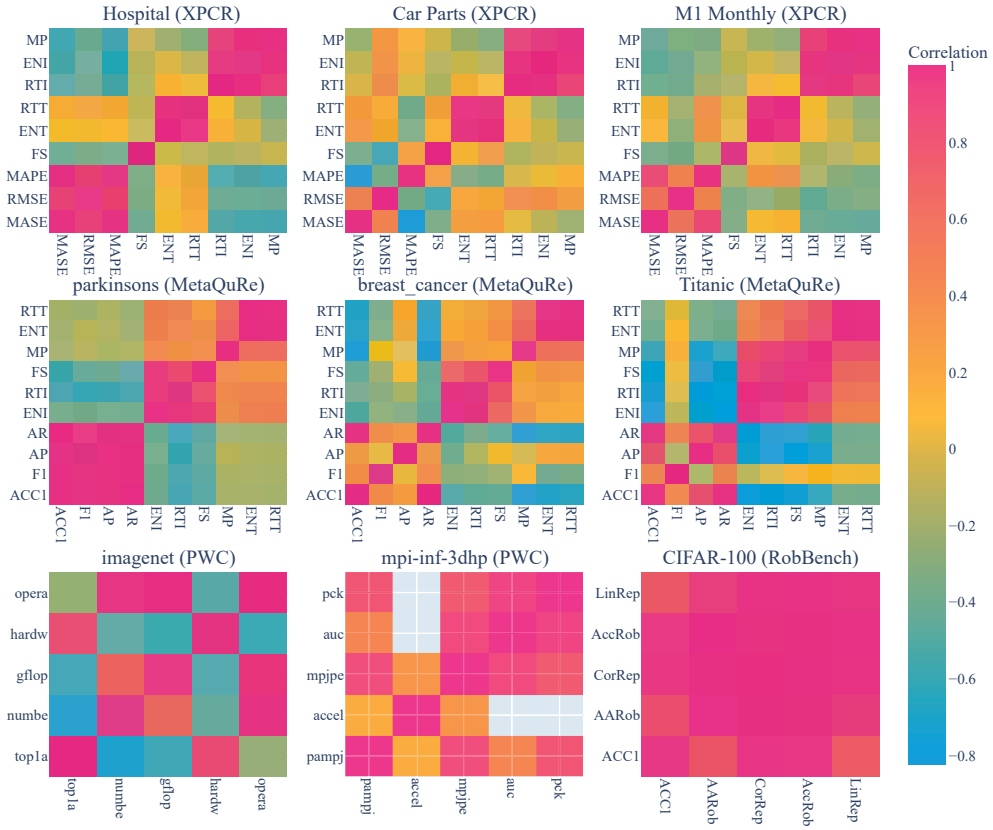

Figure 3.14: Correlation matrices across different datasets. In the first and second row, a fixed pool of models was tested, showcasing that performance and correlations are affected by the dataset choice. In the last row, the *PWC* and *RobBench* results allow for less insight due to incompleteness and strong biases toward positive correlation.

there are some deviations between the databases—not only did we investigate slightly different properties in the respective studies [Fis+22; SFB24], but there are also noticeable differences regarding non-positive correlations, which might originate from the different execution environments. Nevertheless, the mean correlations near zero and very high standard deviations demonstrate that these evaluation databases describe model behavior in a well-balanced and unbiased way, showcasing both correlated properties as well as intriguing trade-offs.

Having understood how property correlation and database biases can be assessed in practice, we can now leave the `ImageNet` classification task and move on to exploring other model evaluations. As listed in Table 3.3, `STREP` currently offers a total of seven databases that will be analyzed in the following. For inspecting their contents and biases, Figure 3.14 depicts the property correlation matrices for nine selected datasets. As before, we see a broad range of correlation values, with generally positive linear dependencies within property groups and no or negative correlation between different groups. However, in-

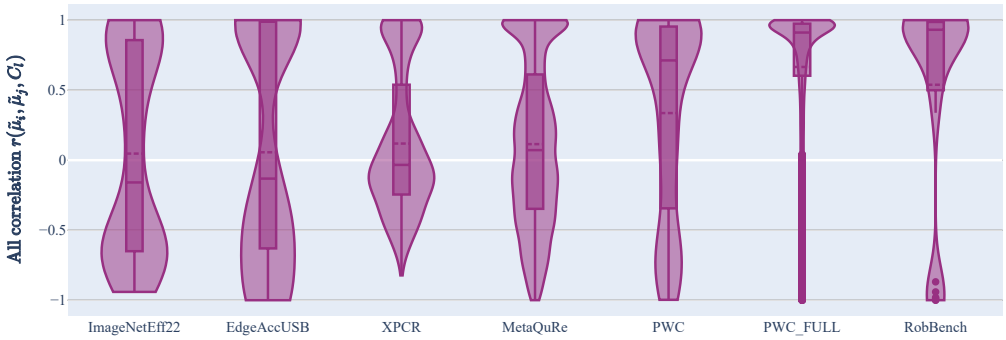

Figure 3.15: Violin plots of all observed correlations in the databases, across all tasks and datasets. From left to right, it reveals the databases to be more biased toward entailing strongly correlated model performance results.

specting the right column matrices of *XPCR* and *MetaQuRe* unveils the positive correlation of quality properties to be much weaker. Moreover, there is evidence that trades can also actually occur within a group of properties, as observable for MAPE and MASE on `Car Parts`. For these databases, the noticeable row-wise differences between the matrices demonstrate how the performances of a fixed set of models is significantly impacted by the data used for evaluation. This empirically evidences NFL [WM97] and also supports the conceptual idea of meta-learning how ML methods and AI models behave across different configurations (see also Section 2.2.2 and Chapter 5).

Looking at the *PWC* results (lower left and middle of Figure 3.14), we encounter completely new properties—they are not introduced in Table 2.1, because the platform does not provide any information on what they describe or how they are computed (except for the name, the σ information encoded via `is_loss`, an optional description that is mostly missing, and the link to the in-depth paper). In addition, we encounter cases where the correlation between certain properties cannot be calculated (middle plot), because the necessary performance information was not reported for enough models (computing Equation (3.7) requires assessed properties for a minimum of two models). While some interesting correlations are given in the left matrix, we found more investigated properties for the manually benchmarked `ImageNet` databases. The middle matrix only depicts positive correlation, an effect that is even more pronounced in *RobBench* (right plot). These results indicate that no proper trade-offs were investigated when the models were originally evaluated. This is further evidenced in Figure 3.15, which depicts violin plots for all evaluation databases. The individual distributions of correlation values were calculated across all investigated learning tasks, datasets, and environments. The custom assembled databases have a mean correlation near zero and stretched out hourglass shapes, however *PWC* and *RobBench* seem to be biased toward reporting on mostly positively correlated properties (high mean value and smaller box plots). In this comparison, it can also be seen that the smaller *PWC* subset is significantly more balanced (i.e., less biased), as it only features the largest evaluation tables with many different investigated models and properties.

3.5 Conclusion

To end this chapter, let me shortly summarize the contents and connections to the rest of this thesis. As motivated in Section 1.1, transparency and decision making are central problems in the context of AI sustainability. For addressing them, this chapter analyzed the current state of reporting on AI and identified flaws regarding comprehensibility, resource-awareness, and interactiveness. To overcome these challenges, it then presented the methodological framework of STREP, discussed how the theoretical formalization is practically implemented in software, and presented in-depth practical investigations for multi-dimensional model performance, efficiency of execution environments, and biases in reporting. The chapter was based on the following publications:

Raphael Fischer, Thomas Liebig, and Katharina Morik. "Towards More Sustainable and Trustworthy Reporting in Machine Learning". In: *Data Mining and Knowledge Discovery* (2024). ISSN: 1573-756X. DOI: 10.1007/s10618-024-01020-3

Raphael Fischer et al. "A Unified Framework for Assessing Energy Efficiency of Machine Learning". In: *Machine Learning and Principles and Practice of Knowledge Discovery in Databases*. 2022, pp. 39–54. DOI: 10.1007/978-3-031-23618-1_3

Alexander van der Staay, Raphael Fischer, and Sebastian Buschjäger. "Stress-Testing USB Accelerators for Efficient Edge Inference". In: *Proceedings of the 9th Symposium on Edge Computing (SEC)*. 2024, pp. 1–14. DOI: 10.1109/SEC62691.2024.00015

To re-iterate, STREP entails five steps for making reporting more sustainable and trustworthy. Index scaling allows to unify and compare the performance of AI models along the introduced characterization, compound scoring and categorical rating enable non-expert users to understand model performances under consideration of their priorities, and property correlations can indicate biases in report databases. The corresponding software puts these methods into practice and allows to investigate the evaluation databases collected during my doctorate, as presented in this chapter. The analysis revealed intricate trade-offs between quality and resource usage of models and execution environments in the CV domain. Acknowledging the Pareto front of model choices, it became clear that optimality is always subject to user preferences. STREP facilitated relative comparisons, allowing to better explore the performance of models with different configurations. We also saw that some established report databases fail at adequately considering the aspect of resource efficiency, overly focusing on strongly correlated performance aspects.

In short, this chapter represents a prime contribution for advancing AI sustainability, equipping practitioners with methods and software to understand and report model performance in a multi-dimensional, resource-aware, and comprehensible way. In the greater context of this thesis, the methods STREP will be of central importance for creating AI labels and sustainable model selection via CML, which will be respectively discussed in Chapter 4 and Chapter 5. The following chapter will thus go into more details as to how AI labels can be practically created, how this relates to the related literature, and how practitioners react to this novel communication format.

4 AI Model Labeling

As discussed in Section 2.1.5, modern AI does not only provide benefits for various applications and use cases, but has moreover become highly accessible. While traditional ML engineering requires extensive knowledge, the capabilities of complex GenAI models can be explored and utilized by a much broader range of practitioners thanks to AIaaS. As established in the beginning of Chapter 3, making informed and sustainable decisions regarding the use of AI however requires comprehensible reporting, for understanding important practical limitations and implications of AI models. Unfortunately, most established reporting formats discussed in Section 3.1 neglect the importance of high-level comprehensibility, which motivated me to explore whether communication and knowledge gaps can be bridged by more abstract *AI labels*, to be introduced in the following.

For conveying important information at a glance and informing less experienced practitioners in particular, Section 4.1 will introduce concepts for constructing *labels* for AI models. This novel form of reporting communicates intricate ML details at an abstract level and was developed over multiple publications [Mor+22; Mor+21; Fis+22; FLM24]. These works where the first that practically applied the concept of consumer labeling systems to AI systems, however related works and extensions will also be discussed. In Section 4.2, the practical feasibility of AI labeling will be evaluated based on findings from a qualitative user study [Fis+25]. Based on the opinions and statements of a diverse group of practitioners, this qualitative analysis also acts as a guideline for refining AI labeling in the future [Sta+25].

4.1 Labeling Concepts

The idea of AI labeling was motivated by the realization that very diverse stakeholders need valid and comprehensible information on ML and AI. ML engineers and AI experts have the necessary skills to develop, debug, evaluate, and understand AI models, possibly with the help of specialized XAI methods (see Section 2.3.1). Application domain experts (e.g., electronic engineers, physicists, biologists), regulatory authorities, or customers that use or are affected by AI products however cannot be expected to first acquire the necessary fundamental knowledge. As motivated in the beginning of Chapter 3, for using AI in sustainable ways, these stakeholders thus require comprehensible reporting and a communication form that goes beyond individual human-machine interaction and instead serves as “a public declaration of a method’s properties” [Mor+22].

Throughout multiple publications, my colleagues and I strove to put this idea into practice, starting with the ML care label concept that will be introduced in Section 4.1.1. Afterwards, the concept will be expanded toward general AI labeling (Section 4.1.2). Examples for these label variants will be provided and discussed in Section 4.1.3. The section will close with discussing how other scientific works have reacted to the idea of labeling, both with regard to our own concepts as well with other approaches (Section 4.1.4).

4.1.1 ML Care Labels

The concept of ML *care labels* was originally developed in analogy to textile care labels [Mor+22], which only convey the most important practical information to consumers (i.e., users) [Eur24b], As such, AI care labels were envisioned as a graphical communication form that certifies ML methods and their implementations for practical use.

Design Overview

Compared to the already manifested STREP methodology for reporting on AI models, care labels were initially designed to explicitly inform on both theoretical as well as practical properties of ML methods, which were later referred to as *static* and *dynamic* properties [Mor+21]. For that purpose, the initial design, depicted in Figure 4.1, features two segments: The upper left part describes important aspects of theory, while the remainder can be switched out, featuring practical properties for the respective method implementation and execution environment. This neatly connects with Section 2.1.4, which described how theoretic properties of methods might not hold during practical implementation. It is also important to note that the care label study describes methods to consist of specific *components* like training procedures and logic algorithms [Mor+22], thus referring to hyperparameters that impact how the ML method internally operates (see also Section 2.2.1). Models accordingly are obtained from practically performing the method with a specific selection of components in a given implementation environment, which not only results in fixed parameters but also provides information for the attachable part of the label. The colors were chosen in analogy to traffic light systems (green to red), while the hexagonal shapes were inspired by the corporate design of the *Competence Center Machine Learning Rhine-Ruhr*[8], the predecessor project of the *Lamarr Institute*.

Static Properties

Looking at the displayed theoretical properties on the label, one can see that they do not directly relate to the properties defined in Section 2.1.4 (except for the implementation resources at the label bottom). Instead, the labels were conceptualized to inform on more abstract *categories* of properties exhibited by ML methods, with regard to expressivity,

[8] https://www.ml2r.de/

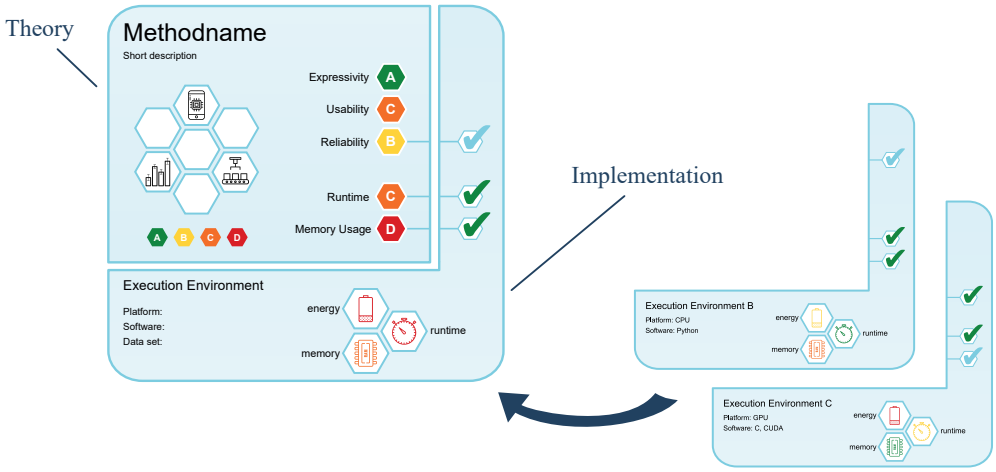

Figure 4.1: Initial design of care labels for ML methods, featuring theoretical properties and corresponding practical information for a given implementation [Mor+22].

usability, and reliability. Expressiveness relates to the method's learning capabilities, for example in terms of its suitability to model complex functions and accompany prediction outputs with additional insights like uncertainty information. Usability encompasses for example the method's sensitivity toward hyperparameters, which in some cases are harder to fine-tune for achieving good performance on given data. Reliability describes how firmly the method in question is based in theory, which for example could relate to mathematical proofs, theoretical guarantees, and error bounds. The central benefit of assessing these more abstract aspects lies in generalization, as different ML methods can be compared along these three overarching categories. In addition to the three categories, static information on the running time and memory requirement is given, based on theory that describes the worst-case resource consumption in the form of big $\mathcal{O}$ notation.

As the categories are not relating to measurable properties of AI models, the respective ratings cannot be derived with the STREP methods introduced in Section 3.2. Instead, the ratings displayed on care labels are obtained from a *criteria* checklist, with each criterion relating to one of the three aforementioned categories. As such, experts are required to assess which important criteria are applicable for the various existing ML methods, leading to an expert knowledge database. The amount of fulfilled criteria for the given method can then be communicated via categorical ratings, as formalized in Section 3.2.4. Naturally, with different customizable components, the criteria assignment can shift, for example calculation kernels can be used to generalize SVMs for also modeling non-linear data relations. The extension to the original care label paper refined the rating process for static properties accordingly [Mor+21]. Taken from this work, Figure 4.2 schematically visualizes how criteria (rectangular boxes) might apply to the given method, some of its components, or the specifically parametrized and trained model. The criteria are associated with the overarching categories and aggregated into discrete ratings via rules. In addition, the labels were conceptualized to feature so-called *badges*, which more abstractly represent

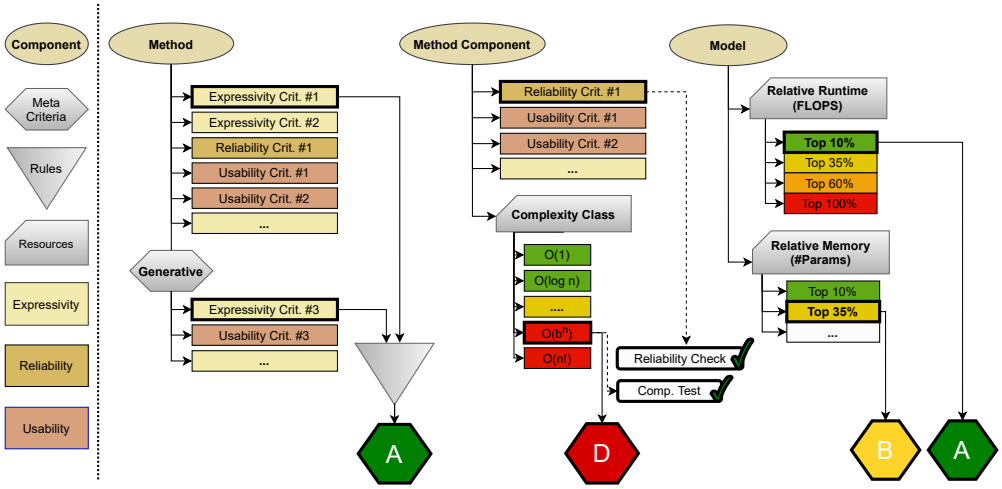

Figure 4.2: Schematic visualization describing how criteria on the method, component, and model levels are together summarized to obtain ratings of categories [Mor+21].

whether a method at hand fulfills noteworthy criteria. As already shown in Figure 4.1, this could for example highlight suitability for stream mining (conveyor belt badge) or availability of uncertainty information (bars with error whiskers).

Dynamic Properties

In the implementation part of the label, check marks denote whether the practical properties of the implemented model align with the theoretical information on the method. For example, to test the big $\mathcal{O}$ notation for running time, the implemented method can be practically applied to multiple datasets with different sizes, in order to check whether it scales as expected from theory. If there are other bounds and guarantees in the context of reliability, they can also be tested in practice. In addition, the care labels feature practical resource information that can be assessed as described in Section 2.1.4 and Section 2.3.3. This information is transformed onto categorical ratings via calculating quantiles over the investigated models, which later inspired the respective `STREP` methodology (Section 3.2.4). High-level information on the utilized hardware and software is also displayed, similar to the characterization described in Section 3.2.1.

4.1.2 Toward Generalized AI Labeling

It is easy to see that the original care label concept focused heavily on describing the theoretical implications of choosing among different ML methods. Applying this concept to a wide range of existing methods and models not only requires immense knowledge about ML theory, but actually explodes in complexity. As Section 2.1.3 discussed, there exists a multitude of ML methods and moreover sophisticated subareas like ensemble learning

Algorithm 4.1 Generating AI labels for a given evaluation database and user objective

Input: evaluation database $\mathfrak{D}$ with performance properties for K models evaluated across L configurations, user-specified objective Ω
Output: Labels for each model evaluation

```
𝔇̃ ← I(𝔇) {index scaling, see Definition 3.2}
𝔇̌ ← Ξ(𝔇̃) {categorical rating, see Definition 3.5}
for k ∈ {1, ..., K} do
  for l ∈ {1, ..., L} do
    if model m_k evaluated on configuration C_l then
      S ← S_{Q,Ω}(m_k, C_l) {compound scoring, see Definition 3.3}
      Š ← ξ(S)
      label_{k,l} = create_label(m_k, C_l, Š, 𝔇, 𝔇̃, 𝔇̌)
    end if
  end for
end for
return {label_{k,l}}
```

and DL, allowing to build powerful models from more basic methods. With this amount of complexity and customizability, the naive approach of differentiating between method components is hardly feasible, as for example the choice of DNN architecture would need to be understood as a single component, whose infinite choices could completely change the method properties. It comes to no surprise, that ML methods, associated criteria, and tests for theoretical properties can only be thoroughly characterized and developed, if at all, "by the research community" in its entirety [Mor+22]. Moreover, all well-performing and publicly available GenAI and AIaaS models belong to the family of DNNs, which arguably renders the comparison of theoretical DL properties somewhat irrelevant.

As a result, and as already mentioned in Section 1.1, the labeling focus thus shifted away from theory and toward acting as practical guidelines for using ML and AI in sustainable ways. The first extension to the care label framework [Mor+21] already put more emphasis on practical properties exhibited by DNN image classifiers, which in literature are usually compared with respect to empirical performance instead of theoretical aspects. As a result, the labels' static categories were adapted to represent important information from the original model proposal (i.e., paper), such as the reported accuracy against which implementations can be tested. In the next adaption, the label design was refined in order to explicitly inform on practical AI energy efficiency [Fis+22]. As such, theoretical aspects were completely discarded, instead focusing on the trade-off between real-world resource consumption and predictive quality, in analogy to the EU energy labels [Eur24a]. With the work on STREP [FLM24], labeling was generalized to report on generic model properties and trade-offs, while being agnostic to the specific application domain or learning task. For the most recent evaluation study [Fis+25], the design was further adjusted to show less parallels to energy labels in order to express its general nature.

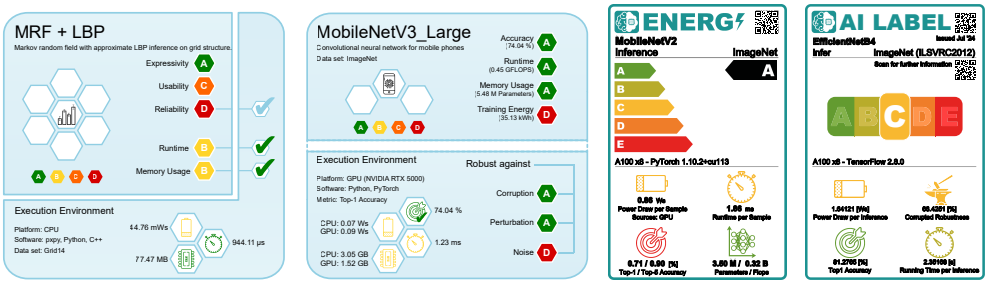

Figure 4.3: Evolution of proposed high-level AI labels (from left to right), starting with the initial care label for MRFs [Mor+22], the adaption for `ImageNet` models [Mor+21], ML energy efficiency labels [Fis+22], and finally, generalized AI labels [FLM24; Fis+25].

Note that discarding theoretical information on labels however does not mean that the labeling procedure becomes trivial—quite the opposite, a rich and sound methodological framework is needed for generalized AI labeling. The `STREP` framework acts a possible candidate and was in-depth presented in Section 3.2. These methods can be combined to generate labels for a given evaluation database, as summarized in Algorithm 4.1. First, the given model properties need to be made comparable via index scaling (Definition 3.2), which unifies performance results across different execution environments. The resulting relative multi-dimensional performance information can then be transformed into a more comprehensible and intuitive representation, by depicting color-coded icons based on categorical rating (Definition 3.5). As last step, the overall model performance and trades among properties or property groups can be interactively unified via compound scoring based on user preferences (Definition 3.3), which can also be communicated as a discretely rated and color-coded visual entity. By also featuring QR codes, labels moreover enable users to directly consult more in-depth information resources like the fundamental paper for the respective model or additional explanations on the labeling procedure. Recall that the `STREP` software presented in Section 3.3 implements Algorithm 4.1 and entails a respective `label_generator` [FLM24].

4.1.3 Label Examples

To illustrate the described evolution of AI labels, Figure 4.3 displays an example for each of the four label versions, as presented in the respective papers.

Care Label V0 [Mor+22]

The left label informs on MRFs, a probabilistic ML method introduced in Section 2.1.3. In this case, it is equipped with the LBP inference algorithm, representing a classic case of a configurable component that could be switched out for other algorithms. This configuration theoretically has a lower resource demand at the cost of reliability, because LBP approximates exact probabilistic inference (which would be much more costly) [Pia19]. As

the amount and complexity of features is still a significant factor in the worst-case resource demands of MRFs, they however do not receive an optimal rating. In the assessment, MRFs were rated to be very expressive because they can theoretically model arbitrary probability distributions. On the downside, the require information about the underlying independence graph, resulting in non-optimal usability. For the investigated implementation (`pxpy`), the models scale as expected from theory (green check marks), however for this choice of inference algorithm no guarantees can be made (washed out blue check mark). The ratings for the measured resources were obtained from fixed thresholds.

Care Label V1 [Mor+21]

In the first care label extension, the design was adapted for informing on the more practical features of four investigated `ImageNet` models. In comparison with the competitor DNNs (`AlexNet` [KSH17], `VGG11` [SZ15], `ResNet18` [He+16a], not shown here), the `MobileNetV3Large` [How+19] model has very good static properties in all categories except training energy demand, based on information from the respective publications. For the given execution environment, the practical accuracy aligns with the static information in the paper (check mark) and the resource consumption is also reasonably low (colors obtained from fixed thresholds). In addition, we evaluated the model robustness regarding corruptions, perturbations, and noise on the input data [HD19], with `MobileNetV3Large` in comparison to the competitors performing well in two out of three scenarios.

Energy Label [Fis+22]

The third label showcases the transition toward AI energy labels, aligning the overall label design with the established EU system [Eur24a]. As such, it moved from four to five discrete rating categories, as displayed via the central color scale. As with the EU design, it uses compound scoring (Definition 3.3) to aggregate the investigated properties into a final rating. However, users can control the weight of properties in order to customize the label to their preferences. The most important properties are featured in the lower half, in addition to some text information on the experiment configuration. A QR code forwards interested viewers to the full paper, in this case the one that introduced `MobileNetV2` [San+18].

General AI Label [Fis+25]

The latest variant displayed in Figure 4.3 is taken from the respective labeling evaluation study, where the design was changed to be less similar to the EU energy labels, demonstrating that the label does not only report on energy efficiency. The label therefore also features a different header and compound score scale, inspired by the Nutri-score design [Org22]. Moreover, a second QR code and issuing date was added for more transparency, as already proposed in the `STREP` paper [FLM24].

4.1.4 Reactions, Adaptions, and Related Works

Several works [PPP24; SRA24] have positively reacted to the proposed care labeling for AI models [Fis+22], such as Genovesi and Mönig, who see the framework as an "excellent tool" for auditing and improving AI systems [GM22]. With a side-comment to package inserts in medicine and pharmaceutics, Hanna et al. acknowledged benefits for providing transparency and accountability [Han+25]. The extension toward energy efficiency was for example positively mentioned in the context of climate change challenges [PPP24] and environmental responsibility [SRA24]. Moreover, it has sparked some adaptions like the repository mining study by Castaño et al., which does not explicitly depict visual labels but uses a similar methodology to assess and compare the efficiency of various *Hugging Face* models [Cas+23]. As a second extension, Durán et al. presented "GAISSALabel" [Dur+24], which encloses a web-based tool to evaluate models and generate respective labels that closely resemble our original AI efficiency labels.

In addition to the proposed labels, several others have envisioned documentation solutions in the context of AI. Naturally, these approaches are closely related to the types of reporting described in Section 3.1, aiming to guide users via high-level information while concealing the technical complexity beneath. For example, with an explicit focus on environmental sustainability, a respective eco-label for software was conceptualized [DR20] and could potentially also be issued for software with AI components. Based on the early idea of "datasheets for datasets" [Geb+21] and the Nutri-scoring framework [Org22], a respective "Dataset Nutrition Label" was developed [Chm+22], which however remains much more complex than its very abstract role model. The aforementioned model cards [Mit+19; Lia+24] and fact sheets [Arn+19; Hin+20; Arn+22] are popular representations for describing practical properties of AI models, however these formats are neither standardized nor comprehensible at a glance—as such, they should not be understood as labeling approaches.

For explicitly addressing non-experts, Seifert et al. conceptually proposed AI consumer labels [SSW19], which unfortunately were never practically implemented or evaluated—nevertheless, this work is conceptually close to our proposed labeling. On an even more abstract level, some works [Sch+23; Wis+24] have investigated whether "trust seals" could be beneficial for improving AI trustworthiness (cf. Section 2.3.1). This approach is inspired by well-known trust labeling systems, which for example are frequently encountered in electronic commerce. However, the effectiveness of high-level trust seals is at question [KSH12]—Kim et al. for example first found high-level labels to not "strongly influence consumers' trust" [KFR08], however a high perceived effectiveness was reported several years later [Kim+16]. Similarly mixed results can be seen in the context of AI: Scharowski et al. have found positive effects from trust labeling, especially in the context of "high-stake scenarios" [Sch+23]. In contrast, Wischnewski et al. report "mixed support for the use of AI seals", stating that study participants strongly demanded verification but did not really understand the respective seals [Wis+24]. It should be noted that these works investigated hypothetical AI trust seals, since there is no such established system yet. These studies

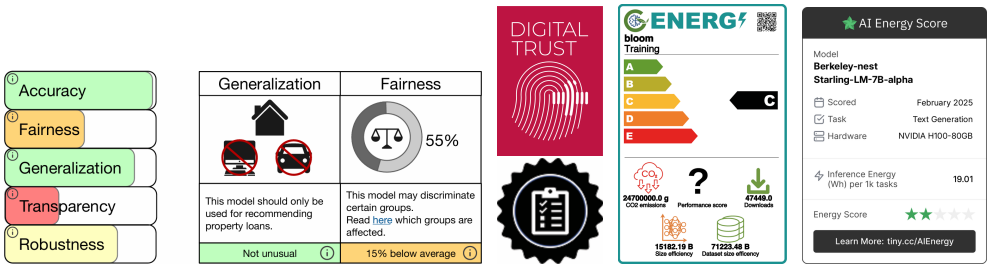

Figure 4.4: Labels proposed or investigated in related works, from left to right: summarizing important AI properties for consumers [SSW19], certifying trustworthiness [Sch+23; Wis+24], informing on AI energy efficiency [Dur+24], and awarding an AI energy score [Luc+25].

also highlighted that the credibility of the seal-issuing institution has an impact on the seal effectiveness [Wis+24], which connects back to literature on general trust types [Mck+11] and trust duality in AIaaS [LS16], as already mentioned in Section 2.3.1.

Recently, the "AI Energy Score" project by Luccioni et al. was launched [Luc+25], closely integrated with the *Hugging Face* platform and conceptually close to our work on labeling energy efficiency [Fis+22]. The initiative comes with an impressive evaluation testbed that even supports proprietary models, however comparing model performance with regard to different hardware setups, training, and predictive capabilities is not (yet) supported—the project purely focuses on measuring the GPU energy consumption when performing inference for a specific task in a fixed execution environment. Like the earlier presented energy labels, the *Hugging Face* label design features information on the tested model, task, evaluation date, and execution environment, as well as the inference energy and the resulting discretized score. The respective leaderboard summarizes efficiency information for a wide range of popular tasks and respective GenAI models, while the methods and concepts discussed in this thesis were mostly applied in specific ML domains. As such, our research papers on the matter might offer fewer practical insights for AI users, however have a richer theoretical foundation for comparing model performance. Further similarities of both approaches can be observed for the overall characterization of AI models and the quantile solution for deriving five-step ratings.

Figure 4.4 gives an overview for some of the labels proposed and used in the mentioned related works. The consumer label (left) only represents a conceptual example, which does not reflect any real ML results [SSW19]. The displayed trust seals are also of hypothetical nature and represent very high-level certificates [Sch+23; Wis+24], making them very different from the labels proposed in this thesis. The "GAISSALabel" [Dur+24] and "AI Energy Score" [Luc+25] are more in-depth and moreover show significant parallels to our discussed work on assessing and labeling energy efficiency [Fis+22]. As such, these work underline the importance and innovativeness of my own contributions toward AI labeling. However, in order to provide maximum benefits for comprehensibility, transparency, and sustainability, AI labels still require careful evaluation with regard to practicability and limitations.

4.2 Evaluation of Labeling Practices

The introduced concepts for AI labels raise the question whether practitioners in the field actually perceive them as beneficial. Fellow researchers made positive comments that indicate a supportive stance, however practitioner feedback is important for validation and refinement. To qualitatively evaluate labeling and identify good practices, we next explore results from our recent interdisciplinary user study [Fis+25]. Concisely, we investigated benefits and limitations by performing inductive coding and thematic analysis of statements encountered in practitioner interviews. The following will introduce the study methodology, present interviewee positions and responses, and close with a detailed discussion of recommended labeling practices. Please note that this section is based on the preprint version from January 2025—by now, the manuscript was revised for publication at the *Conference on AI, Ethics, and Society*, resulting in minor deviations [Fis+25].

4.2.1 Evaluation Methodology

The planning of our evaluation study began with formalizing four central research questions that arose when discussing the potential of labeling [Fis+25]:

RQ1: Who is interested in AI labeling and what are their problems with using or developing AI?

RQ2: What are the practical benefits and limitations of labeling AI model behavior?

RQ3: How are AI labels perceived in comparison to other forms of reporting?

RQ4: How do AI labels and the corresponding certifying authority affect the trustworthiness of AI systems?

RQ3 links back to the types of reporting discussed in Section 3.1, while RQ4 brings in the aspect of trustworthiness, as discussed in Section 2.3.1. For finding answers to our questions, we decided to follow a qualitative approach based on thematic analysis of practitioner interviews, for which we obtained ethical approval from the University of Duisburg-Essen Computer Science faculty. Based on a public call and social media campaign, we successfully recruited 16 interviewees from various backgrounds and with vastly different levels of AI experience (self-assessment), as summarized in Table 4.1. Based on our provided orientation help, the *beginner* (1 person) is expected to have a rough idea of but no practical experience with AI, *users* (4) should have at least used AIaaS, *engineers* (8) have gained some experience with performing ML on custom data, and *experts* (3) possess a rich understanding of and extensive practical experience with ML and AI. Our interviewees applied with diverse academic backgrounds, with about half of them having successfully completed a master's or diploma degree. Written consent was obtained before conducting and recording the interviews with *Zoom*, after which each participant was compensated with 15€.

Table 4.1: Jobs and Skills of Study Interviewees [Fis+25]

ID	Job Title & Company Description (Employees)	Gender	Age	AI Skills
I1	AI Manager for an Industrial Manufacturer (5000-10000)	male	—	Engineer
I2	Researcher for a Research Service Provider (51-200)	male	—	Engineer
I3	Student & Software Developer for an IT Service Provider (5000-10000)	male	20	Engineer
I4	Student (40000-45000 students at the university)	male	21	Beginner
I5	Student & Software Developer (self-employed)	male	22	Engineer
I6	Solution Engineer for an IT Service Provider (50-200)	male	28	User
I7	Startup CEO for an AI Service Provider (2-10)	male	29	Expert
I8	Analytics Platform Manager for a Public Service Provider (1000-5000)	male	30	Engineer
I9	Software Developer for a Lottery Service Provider (50-150)	male	31	Expert
I10	Software Developer for an IT Service Provider (self-employed)	male	31	Expert
I11	Data Scientist for an IT & AI Service Provider (11-50)	female	32	Engineer
I12	Researcher for a Telecommunications Provider (201-500)	female	32	User
I13	Development Engineer for an Industrial Manufacturer (5000-10000)	male	43	Engineer
I14	Maintenance Manager for a Public Service Provider (5000-10000)	male	46	User
I15	Principal Cloud Engineer for an IT Service Provider (51-200)	male	47	User
I16	Software Architect for IT Services (self-employed)	male	48	Engineer

We formulated an internal interview guide to structure the interviews into four parts, aligning with our research questions. Interviewees were asked to introduce themselves and describe their daily work with AI, including respective difficulties. Next, they were shown two labels as obtained from `STREP`, from which one is displayed in Figure 4.3 (right). We asked the participants to describe first impressions, draw comparisons between the labels, and discuss relations to their daily AI work (in some cases requiring additional explanations on details of labeling). For the third part, we explored how interviewees use and judge the different types of reporting, showing an overview similar to Figure 3.1. Toward the interview end, we went into an open discussion concerning trustworthiness and the role of AI labels, asking about possible labeling authorities and interviewees' positions toward certification and regulation.

The audio recordings were transcribed via *OpenAI*'s `whisper-large-v3` speech recognition model [Rad+23], deployed locally with the `Shoutout` tool[9]. After manually revising the transcripts, we performed inductive coding to analyze the interviews using `MAXQDA`[10]. We iteratively discussed individually coded interviews and also checked for intercoder agreeability at the end of our analysis, reaching 90% on two interviews. The final code system follows a hierarchical structure, encompassing 136 codes that refer to a total of 1130 text passages. Table 4.2 gives an overview for our code system, summarizing the codes of the top-level families (number of codes and occurrences), and providing respective example quotes [Fis+25]. Supplementary materials for our study, including the interview guide, transcripts, and code system, can be explored at `https://github.com/raphischer/labeling-evaluation`.

[9] `https://github.com/RWTH-TIME/shoutout`
[10] `https://www.maxqda.com/` (Version 24.5)

Table 4.2: Overview of the Derived Code System with Occurrences and Quotes [Fis+25]

Code Family (RQ)	Size	Occ	Quote Examples
General Codes (1)	8	63	"To use AI [...] to counter the shortage of skilled workers"
Types of Daily Work (1)	9	64	"develop an app to detect tolerable products in the supermarket"
AI Use Cases (1)	10	41	"monitoring the machine condition such that we can make predictions"
ML Methods (1)	7	64	"the AI evaluates whether the typed text contains specific data"
ML Tools & Brands (1)	8	36	"I used scikit-learn models and also worked with TensorFlow"
Requirements on AI (1)	13	118	"My boss doesn't care much about the process, he wants results"
Benefits (2)	12	140	"Your label helps me to decide immediately, it saves a lot of time"
Limitations (2)	21	205	"I don't get how the value is included in the overall scoring"
Property Importance (2)	5	64	"the primary objectives: reducing time and enhancing accuracy"
Associations (2)	3	31	"like I'm looking for a washing machine at the DIY store"
Target Audience (2)	1	10	"the addressees are likely to be people who are intensively involved"
Workflows and Use (3)	12	61	"different agendas and newsletters as a regular source of information"
General Comparison (3)	4	46	"It is time-consuming – that is the disadvantage of other approaches"
Who Needs Trust (4)	3	26	"it helps to understand how the model works if you are a developer"
Reasons for Trust (4)	9	91	"if it has a university stamp on it, it seems more trustworthy"
Dimensions of Trust (4)	11	66	"trust in AI, or trust in a label – these are two different things"
Total	136	1130	

4.2.2 Statements and Positions

With the details of our study methodology in mind, answers to RQ1–RQ4 are next formulated based on our thematic analysis. The encountered opinions are visually summarized in Figure 4.5, showcasing the complexity of our code system [Fis+25].

Who Is Interested in AI Labeling and What Are Their Problems With Using or Developing AI? (RQ1)

The diversity of participants as given in Table 4.1 already demonstrates the diversity of practitioners that are interested in AI labeling. To characterize target audiences and their needs further, we investigated general problems and positions, types of daily work, AI use cases, and requirements on AI, as mentioned by the interviewees.

A central problem for example seems to reside in communication, with I1 describing difficulties of getting "employees on board so that they can actually use the new tools" (p. 26), and I11 mentioning issues with "customer communication and expectation management" (p. 50). Many interviewees described business growth thanks to AI, however there were also many insecure statements, for example expressed by I15: "I have quite a few concerns, but on the other hand, I find AI very convenient" (p. 26). In this context, it is important to distinguish "between AI tools that are used during work or AI tools that are incorporated into products" (I12, p. 28). This demonstrates the different roles labeling might have for practitioners, either describing solutions that they work on or resources used during their daily work. With respect to the latter role, our interviewees most frequently described

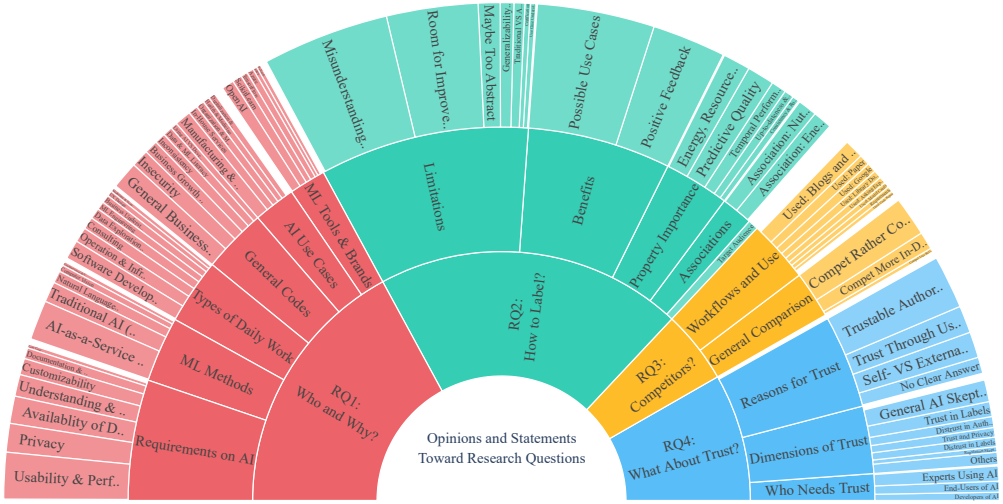

Figure 4.5: Overview for the encountered and analyzed opinions toward labeling, structured by the research questions of our evaluation study [Fis+25].

work relating to software development, infrastructure and operationalization, consulting, as well as data exploration and analysis. Using readily available AIaaS was mentioned more often than performing custom ML engineering, which evidences the recent paradigm shift discussed in Section 2.1.5. Underlining this, I7 stated that "when people talk about AI today, they no longer mean deep learning, they mean solely and exclusively large language models" (i.e., GenAI, p. 4).

The requirements on AI as formulated by the interviewees were found to be as diverse as their daily work and use cases. Usability and performance were especially noteworthy, or as formulated by I5, it is important "whether the result works or whether the AI itself is bad" (I5, p. 98). Most interviewees here expressed the importance of high prediction quality; however, few mentions were also made with respect to consistency, robustness, and response times. Several interviewees made comments on the importance of data protection and privacy, with I11 stating that their company does not even deal "with personal data" for this specific reason (p. 14). Developers also mentioned data availability to be very important, for example I9 argued that "the biggest step in ML and training is data collection, and then, feature engineering and data processing" (p. 40). Lastly, with respect to communicating about specific models, many remarks expressed the importance of understanding and transparency, customizability, and documentation and reporting. I1 for example highlighted the need for "carrying out educational work" for employees (p. 30), I8 demanded an improved "understanding in terms of what is happening, so that we don't get a black box" (p. 8), and I4 mentioned problems with regard to "finding out what is the right model for my area of application" (p. 40).

Coming back to RQ1, we conclude that potential users of AI labels and their demands are extremely diverse. Our findings suggest that labeling could be a means of connecting

different stakeholders, such as experts that develop AI solutions and people that simply use it for their daily work.

What Are the Practical Benefits and Limitations of Labeling AI Model Behavior? (RQ2)

Facing AI labels, our interviewees gave extensive feedback that was divided into describing benefits, limitations, and specific improvement suggestions. The latter two should not be considered as statements against labeling in general, as we specifically asked for critical comments in order to develop practical refinement guidelines. First of all, many positive comments were made with respect to the label complexity and design. Labels were appreciated to be "informative at a glance" (I14 p. 56), "very consumer-friendly" (I3, p. 84), and "optically appealing" (I7, p. 94), with I1 believing them to "help in any case—as there are more and more models, it is increasingly difficult to keep an overview, and the more compact the information is, the better" (p. 100). Thanks to the abstraction of complex information, over 50 comments described benefits for decision processes and "comparing different models with each other and seeing how well they perform" (I2, p. 70). Additional advantages are seen for communication and knowledge transfer (20 mentions), like I7, who saw "greatest added value for customer presentations" (p. 174), and I14, who perceived labels as "an excellent way" for presenting users with the "basic information" they need (p. 170). Further benefits were mentioned with respect to transparency, advertisement, and validation (or certification) of results.

In contrast to these positive remarks, the interviewees however also critically viewed the abstract label design, for example I11 mentioned complex factors around models "which I don't think you can necessarily cover in a label" (p. 164). At the same time, the displayed properties led to confusion and were criticized to be too technical, with I3 stating that they might be "interesting for developers", however questioning whether non-experts would "really understand" them (p. 216). Most misunderstandings occurred with regard to the robustness and accuracy information, where several interviewees could "not really imagine what it means" (I4, p.48). The importance of a "qualitative, subjective" (I7, p. 66) evaluation was also stressed, with I12 demanding better information on the "real experience" of AI users. Further confusion arose with regard to the relative rating and compound scoring of STREP (see Section 3.2), for example I9 requested more transparency for understanding the impacts of properties for deciding "which categories it belongs to" (p. 134). While the color-coding was also positively acknowledged, one interviewee mentioned inaccessibility for color-blind people and others only understood the associated relative rating when facing two labels (I7, p. 78: "I didn't realize before that the icons at the bottom were color-coded"). Participants argued that the label complexity should be aligned with the skills of the target audience, but there was no clear consensus on who labels should address, benefits were mentioned for "people who want to use AI" (I5, p. 228), "decision-makers and customers" (I7, p. 74), or "people who are intensively involved" (i.e., experts, I14, p. 64).

Having explained the interactive aspect of labeling and STREP (cf. Section 3.2.3), interviewees approved of the idea that user's priorities can be incorporated, like I16 who appreciated the "opportunities to demonstrate what is important to the developer" (p. 50). As such, adaptability of the labeling procedure was stated to be beneficial for aligning management expectations with development: "if I knew what my boss wanted, I would go to the website, set the weighting, press enter and then I would pull out your label" (I13, p. 108). The conceptual idea of an interactive dashboard "that exactly displays those things that are relevant to you" (I7, p. 166) was also positively received. The role of labels in the context of model selection (cf. Section 2.2.1) was also discussed, with I13 highlighting the need for guiding users with the use of AI and answering potential questions like "Which model should I use? How do I find my way around?" (p. 72). Here I2 (as well as I5 and I8) saw room for improvement and advocate to explicitly inform on "the combination of model quality and application area" (p. 118).

As an important last aspect, we observed that the previously mentioned requirements (RQ1) on AI did not really align with the interviewee preferences when being asked to rate the importance of label properties. While a "good mix of all points" (I6, p. 226) would be desirable in an ideal world, the predictive quality (i.e., accuracy) and resource consumption (energy) were seen to be most important. It is important to note that the importance of sustainability was only observable after the interviewees faced the labels, with I1 reporting sustainability to be "the central driver of [their] corporate strategy" (p. 54) and I2 advocating a general "frugality of system complexity" (p. 74). The interviewees also correctly identified and discussed the associated trade-off, as "power costs money when [the system] is operationalized" (I13, p. 72), yet "when accuracy is key, I would also have to accept higher power draw" (I15, p. 52). The mismatch of findings regarding AI requirements and importance of properties showcases how unbiased reporting via labels not only enables understanding and transparency, but can also nudge practitioners toward resource-awareness.

Summarizing a response to RQ2, labels can be beneficial for decision processes and knowledge transfer, and moreover hold strong promises for SD. This potential is however hindered by the problem of balancing simplicity and complexity, requiring customization mechanisms for informing specific target audiences.

How Are AI Labels Perceived in Comparison to Other Forms of Reporting? (RQ3)

As already discussed in Section 3.1, labels should not be seen as superior to other reporting types, however the relations between them is an interesting aspect to investigate. In the interviews, we found all types of reporting to be important: While gray literature received the most mentions concerning active use, the depth of academic papers was also positively acknowledged, further evidencing the trade-off between simplicity and complexity. I8 for example mentioned a high popularity of online blogs like *Medium*, likely "because it's often a practical example that is well explained and easy to work with" (p. 116). In contrast, I6 mentioned concerns regarding trust and reliability: "What bothers me about Medium is

that […] anyone can write anything" (p. 182). As additional examples for this reporting type, I13 and I14 praised educational videos for getting started with ML, while I12 uses journalist articles to learn about "trends" and seeing "what others do" (p. 186). In order to gain a deeper understanding, I3 however stated that you "have to look at [the paper], in any case", and I7 described how he regularly scans them for comparing "benchmarks and other models" (p. 114).

In comparison of AI labels with other reporting forms, many mentioned benefits for having fast access to information, like I9, who saw the amount time needed for reading as "the disadvantage of all other approaches" (p. 219). As other reports feature "significantly more text [and] significantly more data" (I3, p. 168), I8 believes that "there is quite a lot of complexity involved and I would first have to have a pretty good understanding of it" (p. 104). With a well-established labeling system, I7 imagined strong benefits for quickly learning about models, as "you don't even have to read [the sources] anymore—you just know how good [the models] are." (p. 110). In contrast, the other types of reporting were stated to be "particularly relevant when depth is needed" (I16, p. 86), with I3 stating that when you: "read a paper about an AI, I can probably understand it much better than if I just look at the label" (p. 172). While "none of these [forms of reporting] are as easy to understand as the label" (I3, p. 168), I2 correctly stated that "we need them all—the difference is, which target groups do I face?" (p. 110), supporting the earlier discussion around Table 3.1.

Answering RQ3, we conclude that labels complement other reporting types, allowing for faster information access but not satisfying practitioners who require deeper insights.

How Do AI Labels and the Corresponding Certifying Authority Affect the Trustworthiness of AI Systems? (RQ4)

Lastly, the aspect of trustworthiness in the context of AI labels was investigated, for which we identified two prime perspectives. I11 brought it the point, stating that one has to differentiate between "trust in AI [models], and trust in a label [that summarizes model performance]" (p. 152). In addition to these two aspects, we also encountered comments regarding general AI skepticism and regulation skepticism, with I1 encountering various views on AI in the company, "from euphorically enthusiastic to rather skeptically rejecting" (p. 30).

Most interview comments regarding trust in labeling were positive, such as I13, who believes in label benefits "because it's approved by professionals and trust is created" (p. 52). On the other hand, I7 was unsure whether he can "rely on such a label" (p. 82), stating that he would rather "test [AI models] himself" (p. 82), and I11 doubted that "performance parameters help with a question of trust" (p. 164). Large parts of the discussions revolved around suitable labeling authorities, which further evidences the importance of institutions in the context of trustworthiness [Wis+24]. Our interview participants advocated the idea of having labels produced by a "central" (I8, p. 124), "official" (I14, p. 146), and "independent"

(I15, p. 122) authority, yet did not agree on a concise answer on who would be a good candidate for that position. Moreover, nearly half of the interviewees voiced concerns regarding subjectivity and feared that authorities could be "bribed" (I4, p- 160). Discussing risks of manipulation and tricking the labeling system, references to incidents with organic labels were made (I8, p. 124). As a possible solution, I9 argued, that, "quite democratically, the users" could be involved in the labeling procedure (p. 227). The pros and cons of self-certification versus third-party involvement were very actively discussed—I3 stated that "there must be something centralized, such that not everyone is allowed to make up their own label" (p. 184), however others argued that access to the labeling procedure "creates transparency and you can check [...] if it works as I imagine it will" (I5, p. 252). In contrast to the importance of authorities, I8 stated that "what works well, what is well explained, counts more than who published it" (p. 116), and I4 also formulated performance as a means to increase trust: "most people probably don't care [about understanding the system]. The main thing is that the end result is correct" (p. 180).

Moreover, the question of whether labeling can increase trust in AI once again requires to differentiate between different audiences, as interviewees anticipated different trust requirements depending on the respective AI proficiency. I11 for example stated that "as a user of an AI, then of course I have the least trust" (p. 186)—in this context, labels can have a twofold effect. While end-users might be discouraged by the rather technical properties (I15, p. 106, I1, p. 96), there were also positive remarks that the model performance could become more tangible and understandable (I6, p. 210). Concerning developers, I11 stated that "I don't have a problem with trust in the sense that I'm the person who decides what kind of model to use" (I11, p. 178), demonstrating how this target audience likely has less trust requirements. In contrast, I8 explained that many developers require and implement use-case-specific XAI methods for trusting their model predictions.

Coming back to RQ4, our results once more evidence the complexity of advancing trustworthy AI—responsible authorities and personal experience are the biggest factors for improving trust, and labels can be a crucial element for connecting both.

4.2.3 Discussion and Recommended Practices

The investigations around our research questions revealed four central themes concerning AI labeling [Fis+25], which also allow for formulating actionable guidelines [Sta+25].

First of all, the tension between offering simplicity and detailed information was centrally discussed in the context of the label design. Thanks to their level of abstraction, interviewees saw strong benefits for making and communicating decisions on AI use, however also voiced concerns regarding loss of important details and performance nuances. Formulating a first takeaway, "**labels must strike a balance between providing an overview that is both accessible and meaningful without sacrificing important detail**". Navigating this balance requires to characterize the "desiderata of the various classes of stakeholders", s it has been manifested with regard to AI explainability [Lan+21].

If labels do not adequately meet these demands, they could even negatively impact AI understanding, and potentially, trust. However, means for interaction and customization can help in shaping labels toward user expectations and linking them to other related reporting formats [FLM24], such as in-depth papers with practical implementation details.

As the second point of discussion, our study unveiled that AI labels do not only convey information but can also nudge users and influence their decision. As such, one has to acknowledge that "**labels do more than simply present information—they actively shape the decision-making process by highlighting the factors deemed most important**". From a psychological viewpoint, labels do not only resemble signals that positively affect trustworthiness, but can also be interpreted as persuasive cues [Fis+25]. As such, they can potentially shift the questionable values of the ML community [Bir+22] toward environmental awareness [Lia+24] and sustainability (see Section 2.3.2).

Next, it is also important to note that the ambivalent relationship between labeling and trustworthiness extends far beyond signaling theory. While serving as a helpful tool for fast decision making and getting started, our interviewed experts questioned whether labels could genuinely eliminate the need for testing model behavior in practice. Moreover, when learning about AI models with labels, users might overlook important details or misinterpret the displayed information, potentially resulting in inappropriate use and distrust [WKM23]. Our results also underlined the importance of finding a trustworthy labeling authority, however interviewees did not agree on who could be a good candidate and actively discussed advantages and problems of third-party labeling in comparison to a community-driven approach. We therefore argue that "**labels need to be seen as part of a larger trust building process that involves transparency, verifiability, and user experience**". The discussions around trustworthiness and suitable labeling authorities connect to other studies [Wis+24] and might possibly by rooted in organizational distrust as a by-product of "surveillance capitalism" [Sæt21; Zub18].

As the last point, we encountered many different perspectives and views on labeling, which are naturally linked to the striking diversity of our pool of participants. We deduce that AI proficiency can change the position toward labeling and thus might have acted as a moderating effect in our study [Fis+25]. As it has been already established for AIaaS [Bre+23], this moreover implies that labeling should not be considered as a "one-size-fits-all" solution and has to remain adaptable to the various needs of AI practitioners. Instead, "**labels must allow for customization, ensuring that different audiences can extract the information they need**". This also connects to studies on trust seals [Kim+16], where characteristics of consumers were discussed to impact the effectiveness of sealing. Conversely, fully leveraging the power of AI labeling necessitates to understand user characteristics and offer a personalized labeling experience.

From the original care labels [Mor+22] to the generalized AI labels discussed here [Fis+25], the concepts have already evolved a lot. Our analysis evaluated labeling and formulated practices for future refinement, which were later transformed into actionable design principles [Sta+25]. Based on these results, labels can be further improved to best serve their purpose of connecting ML experts with various AI stakeholders.

4.3 Conclusion

Before this chapter comes to an end, let me briefly re-iterate the concepts and practical findings for AI labeling and explain how they connect to the greater thesis context. Knowledge and communication gaps are a central challenge in ML engineering and become even more prevalent with the shift toward AIaaS. Building on the STREP methodology (Section 3.2), this chapter focused on establishing a means for bridging these gaps and informing non-experts about important ML trade-offs. For this goal, it conceptually introduced and practically evaluated AI labels, and moreover formulated guidelines for future adaptions, all in all representing the second contribution of this thesis. The chapter was based on the following scientific publications:

Katharina Morik et al. "Yes We Care! - Certification for Machine Learning Methods Through the Care Label Framework". In: *Frontiers in Artificial Intelligence* (2022). DOI: 10.3389/frai.2022.975029

Katharina Morik et al. *The Care Label Concept: A Certification Suite for Trustworthy and Resource-Aware Machine Learning*. 2021. URL: https://arxiv.org/abs/2106.00512

Raphael Fischer et al. "Bridging the Communication Gap: Evaluating AI Labeling Practices for Trustworthy AI Development". In: *Proceedings of the 2025 AAAI/ACM Conference on AI, Ethics, and Society*. (forthcoming). 2025. URL: https://arxiv.org/abs/2501.11909

The contents of these research works are of high novelty and were well-received, not only for formulating the idea, but also for putting it into practice. Closely tied to the STREP methods, we explored how complicated ML properties can be communicated in a comprehensible and quickly consumable way. While the original care labels were also envisioned to inform on theoretical ML aspects, the refined general AI labels put the focus on practical model properties that are more relevant for non-expert users. At the time of writing, a wide range of ML model evaluations offered with the STREP software can already be labeled and explored. Moreover, users can generate labels from their custom evaluation databases. The first practical evaluation study on AI labeling found strong potentials for knowledge transfer and decision making, however also discussed issues related to target audience diversity, technical understanding, and the question of trustworthiness. The recommended practices allow to further refine labeling under consideration of the simplicity versus complexity balance, the role of labels as nudges, the trust-building process connected to labeling, and the importance of customizability. As such, important steps toward AI labeling have been made and the results and reactions support future extensions.

Looking at role model labeling systems, and drawing from our evaluation study, I believe that AI labels need to become even more abstract—by only featuring ratings for very intuitive property groups, such as resource consumption, predictive quality, or interpretability, the technical confusion can be reduced at the highest level. While depicting less details, the design can also be adapted to be less similar to other labeling systems, in order to receive an unbiased reaction from users. Practitioners that want to dive deeper can easily

do so by following the QR code, which forwards them to the reporting framework with additional information and more in-depth reporting formats. As a next step, this framework could also be extended toward offering an easy access point for interactive AI model recommendations. Users would formulate their requirements and priorities and instantly receive a high-level summary of applicable models, alongside respective labels and options for further details [Sta+25].

To conclude, AI labeling can provide users who lack fundamental ML knowledge with a novel communication format that (a) makes the intricate behavior of AI models more transparent, and (b) allows them to make informed decisions. As such, labels directly address the two central issues that motivate this thesis and hence should be considered a powerful lever for SD in the context of AI. Reflecting on the thesis contributions discussed so far, there was a strong focus on the sustainable *use* of AI models. We have yet to understand how sustainability aspects can be best considered during model *development*, which directly leads into the next chapter.

5 Sustainable Model Selection via Meta-Learning

Until now, this thesis has presented contributions for analyzing and comparing the behavior and properties of already trained ML models. The proposed methods for better reporting (Chapter 3) and comprehensible communication via labeling (Chapter 4) enable practitioners to better understand the implications of deploying models, thus empowering them to make sustainable decisions when using AI. However, an important question has not yet been addressed: How can one be sustainable during AI development, or in other words, when looking for a suitable model to train on a specific learning task and dataset? As highlighted at the start of Section 2.2, only relying on pre-trained GenAI models is not advisable when dealing with domain-specific problems. Available multi-task models might provide outputs for given data, however were not optimized for the custom scenario, making them resource-heavy and opaque regarding their internal workings. As long as there is some available learning data about the problem, it therefore makes more sense to develop and train specialized models, which is also recommended in the context of environmental sustainability [LJS24]. However, considering the vast space of possible options introduced in Section 2.1.3, practitioners require guidance in order to make sustainable and informed decisions during AI development.

As discussed in Definition 2.7, the problem of selecting suitable models to train on given data fueled the idea of AutoML and has brought forth many impressive methods and frameworks. However, as mentioned in the final remarks of Section 2.2, these approaches often do not sufficiently acknowledge the importance of sustainability aspects [Tor+23]. Answering the call for sustainable and green AutoML, this chapter contributes the novel CML approach, which is schematically visualized in Figure 5.1 and will be formalized in Section 5.1. As an extension to the basic meta-learning idea, it allows to consider and learn multiple aspects of model performance, resulting in two benefits: Not only can user preferences be considered during the model selection, but the meta-learners moreover accompany every individual selection decision with multi-level explanations. The CML approach was distilled from the methodologies of two publications, which demonstrate practical feasibility on vastly different application scenarios. Accordingly, Section 5.2 will explore how CML can be utilized to select well-performing DNNs in the challenging domain of time series forecasting [FS24], balancing the importance of predictive performance, model complexity, and resource consumption. In Section 5.3, CML will be applied to recommend ML methods for tabular data, with a stronger focus on exploring how the choice of execution environment impacts model selection [Fis+24].

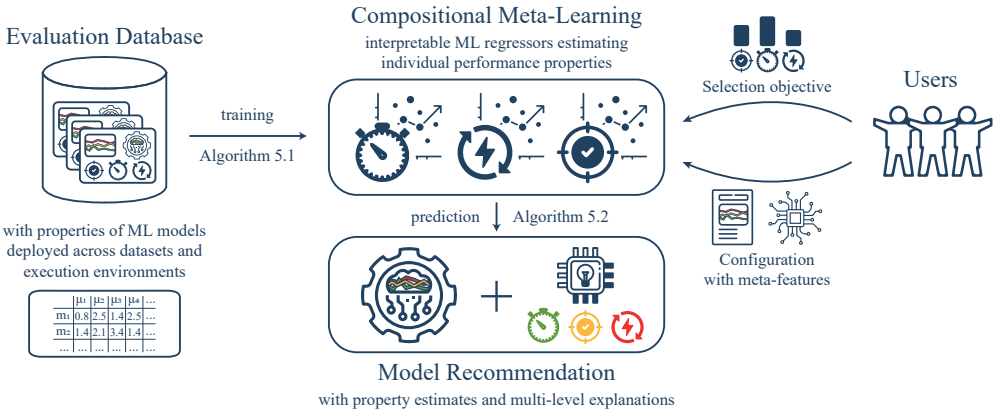

Figure 5.1: Schematic overview of how CML enables sustainable model selection, adapted from the visualization in the first associated publication [FS24].

5.1 Methodology

A conceptual overview for CML is given in Figure 5.1, showcasing the most important elements. Starting on the left, performing CML first of all requires access to empirical model performance results across various datasets, stored in a respective database. Relating the properties of evaluated AI models to the properties of datasets paves the way for model selection via meta-learning. Facing a new dataset, CML first estimates the multi-dimensional performance of all training candidates, based on individual meta-learners that were trained on the database (i.e., following a *compositional* approach). For obtaining a recommendation on which model to train for the given data, the candidates' estimated performances are aggregated under consideration of user-specified priorities, which describe the importance of the different performance aspects. These steps will be further formalized in the following, based on the papers that substantiate this chapter [FS24; Fis+24] and building upon some of the definitions in Chapter 3.

5.1.1 Multi-Objective Model Selection

In its basic form, model selection aims to find the best model with respect to some configuration C (i.e., dataset and environment) and performance property function μ_i, or in other words, solve $m^*_C = \arg\min_{m \in \mathcal{M}_T} \mu_i(m, C)^{\sigma_i}$, as already given in Definition 2.7. It is important to recall that the general search space of applicable models $\mathcal{M}_T$ in practice is often limited to subspaces, as explained in Section 2.2.1. Most AutoML frameworks search in subspaces for specific model types, like single ML algorithms with suitable hyperparameters [Tho+13], ensembles [Eri+20; Feu+22], or DNNs [JSH19; ZLH21]. However, these search spaces are still infinite and thus costly to explore. CML on the other hand was designed to be resource-efficient, and therefore selects models from a finite set of

candidates with fixed hyperparameters. As such, m_C^* in the following will be selected from a model pool $\mathcal{P}_T \subset \mathcal{M}_T$ with K candidates, i.e., $\mathcal{P}_T = \{m_k\}_{k=1}^K$.

Besides the complex search space, a second limitation of the earlier formalized model selection optimization problem is the focus on one specific performance aspect. To be sustainable, the field should acknowledge the multi-dimensional nature of model performance, with candidates being expected to trade certain performance aspects against others. Fortunately, Section 3.2 already introduced formalizations for understanding model performance in multi-dimensional ways, which allows us to understand model selection as a *multi-objective optimization problem*:

Definition 5.1. *Let $\mathcal{P}_T$ be a discrete set of models evaluated for a specific configuration C. In addition, let $S_\Omega(m, C)$ be the compound score of any model $m \in \mathcal{P}_T$ across multiple index-scaled performance properties $\tilde{\boldsymbol{\mu}} = (\tilde{\mu}_i)_{i \in \mathbb{P}_T}$, based on user priorities encoded in the given objective $\Omega = (\omega_i)_{i \in \mathbb{P}_T}$. Optimal model choices m_C^* are then defined as*

$$m_C^* \in \{\arg\max_{m \in \mathcal{P}_T} S_\Omega(m, C)\}, \text{ with } S_\Omega(m, C) = \sum_{i \in \mathbb{P}_T} \omega_i \cdot \tilde{\mu}_i(m, C) \tag{5.1}$$

Note that this formalization of multi-objective model selection is based on the earlier introduced STREP methodology for index scaling (Definition 3.2) and compound scoring (see Definition 3.3), allowing to unify and aggregate different performance properties. Moreover, this approach makes the selection subject to controllable weights ω_i, which can be used to align the importance of performance criteria with user preferences, for example favoring fast, accurate, or low-energy models. Note also that in comparison to the earlier understanding of model selection, Definition 5.1 indicates that there could be multiple optimal models, or in other words, it describes the respective Pareto front mentioned in Section 2.1.4 and Section 3.4.1.

5.1.2 Compositional Meta-Learning

A remaining issue of our new multi-objective model selection formalization is the fact that calculating $S_\Omega(m, C)$ requires to first train and evaluate the model m on configuration C, obtaining performance properties $\boldsymbol{\mu}(m, C)$ that are then index-scaled and aggregated. While optimal models m_C^* could be trivially found via an exhaustive search over the pool $\mathcal{P}_T$, the training and evaluation of all candidates remains very costly. As an approximation, a random search over $\mathcal{P}_T$ could be conducted for only testing a few candidates, which however might be inefficient. However, leveraging the concept of meta-learning [Sch87] in a *compositional* way allows for a more sophisticated approach:

Definition 5.2. *Let $\mathcal{P}_T$ and C respectively be a given model pool and a learning configuration. Let further $X_C \in \mathbb{R}^n$ be a vector of n-dimensional meta-features that encode important information about C, in the form of numerical or categorical variables. Searching for optimal*

models $m_C^* \in \mathcal{P}_T$ can then be approximated via CML, using a surrogate task that instead obtains $\widehat{m_C^*} \in \mathcal{P}_T$ and is defined as:

$$\widehat{m_C^*} \in \{\arg\max_{m \in \mathcal{P}_T} \hat{S}_\Omega(m, X_C)\} = \{\arg\max_{m \in \mathcal{P}_T} \sum_{i \in \mathbb{P}_T} \omega_i \cdot \hat{\tilde{\mu}}_i(m, X_C)\} \tag{5.2}$$

With this approach, the performance of all candidate models m is estimated via meta-learners that were trained to make predictions for index-scaled properties, denoted as $\hat{\tilde{\mu}}_i(m, X_C)$. To clarify this further, instead of evaluating the model m on C, the given configuration is represented by meta-features X_C and fed to simple regression models, which output estimates for the individual performance aspects. The central advantage of this approach is that the prediction via meta-regressors requires much less resources than practically training and evaluating the candidate. Reversing Equation (3.2) allows to also investigate the estimated real-valued properties $\hat{\mu}_i(m, X_C)$, i.e.:

$$\hat{\mu}_i(m, C) = \frac{\mu_i(m^*, C)}{\hat{\tilde{\mu}}_i(m, X_C)^{1/\sigma_i}} \tag{5.3}$$

Based on the estimated performance properties and user-specified weights ω_i, an estimated compound score $\hat{S}_\Omega(m, X_C)$ can be calculated. By making predictions for all model candidates and comparing their compound score estimates, an educated guess about optimal models can be made, identifying $\widehat{m_C^*}$. As a next step, the most promising model(s) can be evaluated on C to assess the true properties, with a good chance that it actually is a good or even the best model. Note that the estimated properties can obviously be prone to errors and thus $\widehat{m_C^*}$ does not necessarily represent the optimal model; nevertheless, the CML approach remains more reasonable than just randomly testing models in the pool.

5.1.3 Training the Meta-Learners

Putting CML into practice requires to train meta-regressors for the individual properties, based on empiric data describing how the models in the pool perform across various learning configurations. This perfectly aligns with the formalization of evaluation databases $\mathfrak{D} = \{(m_k, C_l, \boldsymbol{\mu}(m_k, C_l))\}$, as introduced in Definition 3.1. Such a dataset contains N examples describing how well models m_k perform across configurations C_l with regard to properties $\boldsymbol{\mu}$ and their index-scaled counterparts. Naturally, the respective meta-features X_{C_l} need to also be stored for performing the meta-learning.

It comes to no surprise that identifying good meta-learners for modeling the properties is also subject to NFL [WM97]. That said, it is easy to see why this secondary model selection problem can be approached with an exhaustive search over a fixed set of candidate regression models $\mathcal{R}$: Most importantly, the training data dimensions (i.e., dimensionality of $\mathfrak{D}$) will be rather manageable, as every entry (i.e., row) of measured performance properties represents a single ML experiment performed to evaluate a model candidate on a specific

Algorithm 5.1 Training a CML model selector using STREP methods

Input: Evaluation database $\mathfrak{D}$ with properties $\boldsymbol{\mu}$, Regressor pool $\mathcal{R}$, Folds $F = 5$
Output: Trained meta-learners $\hat{\tilde{\mu}}$ for each property μ

```
D̃ ← I(D) {index scaling, see Definition 3.2}
for each property μ ∈ μ do
    Initialize best MAE: MAE_min ← ∞
    Initialize best model: μ̂̃ ← None
    for each regressor r ∈ R do
        Perform F-fold cross-validation on D̃ with model r predicting property μ̂̃
        Obtain real-valued predictions μ̂ via reverse index scaling
        Compute average MAE: MAE^r_D(μ) ← (1/F) Σ_{f=1}^{F} MAE^r_{D_f}(μ)
        if MAE^r_D(μ) < MAE_min then
            MAE_min ← MAE^r_D(μ)
            μ̂̃ ← r trained on D̃
        end if
    end for
end for
return {μ̂̃ | μ ∈ μ}
```

configuration, at possibly high costs. As such, arbitrarily complex datasets D_T entailed in C will be represented by a much smaller set of meta-features about the data. Moreover, as it is desirable to train a lightweight and interpretable recommendation model [Rud19], the meta-learner model pool should be restricted to basic ML methods that are fast to train and evaluate on small datasets (i.e., $\mathfrak{D}$). This is illustrated by the limited dimensionality of the databases in Table 3.3, from which some will later be used to practically evaluate CML. Simple regression methods, such as linear models and decision trees that are constrained in depth, can be trained on datasets of such sizes within few minutes, obtaining fast and interpretable meta-learners. Note also that these regressors only need to be trained once and can then be used to automatically and efficiently recommend models for any new learning configuration. Therefore, the computational effort of training the CML regressors will quickly be amortized by the high resource efficiency of performing model selection with CML.

With an evaluation database $\mathfrak{D}$ and specified meta-regressor pool $\mathcal{R}$, the CML model selector can be constructed by following Algorithm 5.1. It uses MAE as a standard metric for tabular regression and performs a five-fold cross-validation across $\mathfrak{D}$ to properly validate the meta-learners [HTF09], To get good results on the original scale, Equation (5.3) is used to minimize the MAE on the real-valued properties:

$$\mathrm{MAE}_{\mathfrak{D}}(\mu) = \frac{1}{K \cdot L} \sum_{k=1}^{K} \sum_{l=1}^{L} |\mu(m_k, C_l) - \hat{\mu}(m_k, X_{C_l})| \tag{5.4}$$

Algorithm 5.2 Obtaining a CML model recommendation for a given configuration

Input: Meta-learners $\hat{\tilde{\boldsymbol{\mu}}}$ trained on $\mathfrak{D}$, associated model pool $\mathcal{P}_T$, a given configuration C with meta-features X_C, user objective Ω
Output: Model recommendation $\widehat{m_C^*}$

```
Initialize best estimated compound score: Ŝ*_Ω ← 0
Initialize best model: m̂*_C ← None
for each model candidate m ∈ P_T do
    Estimate model properties μ̃̂(m, X_C)
    Aggregate into estimated compound score Ŝ_Ω(m, X_C)
    if Ŝ_Ω(m, X_C) > Ŝ*_Ω then
        Ŝ*_Ω ← Ŝ_Ω(m, X_C)
        m̂*_C ← m
    end if
end for
return m̂*_C
```

Explicitly considering performance trade-offs can be neglected when only simple and fast meta-regressors are considered. For assessing and comparing the quality of the meta-learners, other metrics can be investigated, like for example the respective error on the index-scaled values, denoted as $\text{MAE}_{\mathfrak{D}}(\tilde{\mu})$, or the compound score error with respect to a specific objective, $\text{MAE}_{\mathfrak{D}}(S_\Omega)$. As the estimates are only used for deciding which model to train, it is also reasonable to check the accuracy of predicting the true best model on the different properties:

$$\text{ACC1}_{\mathfrak{D}}(\tilde{\mu}) = \frac{1}{L}\sum_{l=1}^{L} \mathbf{1}(\underset{m' \in \mathcal{P}_T}{\arg\max}\, \tilde{\mu}(m', C_l) = \underset{m'' \in \mathcal{P}_T}{\arg\max}\, \hat{\tilde{\mu}}(m'', C_l)) \tag{5.5}$$

This metric is very informative because slight estimation errors are neglectable as long as the correct model is selected for training. Similarly, a top-k accuracy can be defined to assess whether the true best model is among the top-k recommendations, e.g., for $k = 3$:

$$\text{ACC3}_{\mathfrak{D}}(\tilde{\mu}) = \frac{1}{L}\sum_{l=1}^{L} \mathbf{1}(\underset{m' \in \mathcal{P}_T}{\arg\max}\, \tilde{\mu}(m', C_l) \in \underset{m'' \in \mathcal{P}_T}{\text{top-3}}\, \hat{\tilde{\mu}}(m'', C_l)) \tag{5.6}$$

Having trained the surrogate meta-regressors on some evaluation database $\mathfrak{D}$, CML can be performed by following Algorithm 5.2. As such, multi-objective model selection (Definition 5.1) for a given configuration C under user preferences Ω can be approximated by (1) making estimates for the index-scaled properties of all model candidates, (2) calculating the respective compound score estimations, (3) sorting the models based on their expected performance, (4) training the best estimated candidate for C. Having trained the

candidate, the real performance of the model can be investigated and compared against the meta-learning estimates. Large deviations indicate that CML has not seen enough comparable configurations during training. As such, the user can decide to test further models from the pool, however is guided by the estimated model performances and as such likely to find a good solution faster than with a random or exhaustive search.

5.1.4 Explainability Aspects

An additional benefit of leveraging meta-learning for automatically choosing models from a discrete set of candidates lies in the by-product explainability, which (as established in Section 2.3.1) is highly relevant for SD. In fact, the proposed CML approach offers explanations on multiple levels, addressing the following questions that users of the system might have:

Q0: How will model m perform on the configuration C with respect to the property μ?

Q1: To which extent do the different properties impact the recommendation?

Q2: To which extent do the meta-features influence the estimates for property μ?

Regarding the first question, the response directly corresponds to the recalculated real-valued estimation for the property (i.e., $\hat{\mu}(m, C)$). It is also straightforward to transform the index-scaled estimates into explanations for answering Q1:

$$E1(m, C) = \left\{ \frac{\varphi(\mu_i, m, X_C)}{\sum_{j \in \mathbb{P}_T} \varphi(\mu_j, m, X_C)} \right\}_{i \in \mathbb{P}_T} \text{with } \varphi(\mu_i, m, X_C) = \frac{\omega_i \hat{\tilde{\mu}}_i(m, X_C)}{\hat{S}_\Omega(m, X_C)} \quad (5.7)$$

In short, for a given property, $\varphi(\mu_i, m, C)$ calculates its contribution (i.e., importance) to the estimated compound score, and the explanation $E1(m, C)$ retrieves the normalized contribution values of all properties. As motivated above, only interpretable models should be used as meta-learners, because they can offer by-product information on feature importance. This directly allows to answer Q2 as:

$$E2(\hat{\mu}) = \left\{ \frac{\beta(x_i, \hat{\mu})}{\sum_{j=1}^{n} \beta(x_j, \hat{\mu})} \right\}_{i=1}^{n} \quad (5.8)$$

β is here used as a generalized function that describes the importance of the meta-feature $x_i \in X_C$ for the meta-learner $\hat{\mu}$ as a non-negative value. Practically, the implementation of Equation (5.8) and β will be subject to the choice of ML method, for example reflecting on the decrease in impurity (decision trees) or feature coefficients (linear regressors) [SJ21]. While this represents a global explanation of the meta-learner $\hat{\mu}$, interpretable

models generally also allow for retrieving local explanations regarding specific outputs $\hat{\mu}(m, X_C)$ [Hol+22].

As a final point, note that changing the user preferences Ω for the compound score estimation does not require to recalculate the individual meta-learner outputs. This makes the model selection highly interactive, which was also argued to be beneficial for understanding the outputs of AI systems and models [Bec+23]. Overall, the discussed aspects show that CML enables users to explore the connections between data characteristics and model performance on multiple levels, and hence should be understood as being explainable by design.

5.1.5 Practical Considerations

Some additional thoughts and remarks are important for putting the theoretical methodology for CML into practice. Naturally, if the evaluation database contains model evaluations across different execution environments, the configuration meta-features should also include respective features, i.e., $X_C = (X_D, X_E)$. In the most basic form, X_E could represent one-hot encoded categorical information, however the meta-features could also entail more informative data, like the available Random-Access Memory (RAM) capacity, number of processor cores, or even the processor thermal design power. Regarding the dataset, CML was purposefully formalized to be agnostic as to how meta-features X_D are extracted, allowing for different approaches like classical data statistics [Dem06] or learned representations [JSG21]. Note that Definition 5.2 does also not specify how the candidate choice m should be considered during meta-learning. In the *MetaQuRe* paper, different meta-learners were trained for each candidate in the pool [Fis+24], however one could also encode information on the available models via respective meta-features X_m [FS24]. It is important to remember that choosing the total number of meta-features should also be considerate of the number of observed model performance results in $\mathfrak{D}$ (i.e., N), in order to not overfit the meta-learners.

Another crucial thing to keep in mind is that all meta-learner outputs should be bounded to the unit scale (like with index scaling, see Definition 3.2), in order to correctly calculate the estimated compound scores. As an alternative approach, one could also train the meta-learners to estimate the real-valued properties and then index-scale the predictions based on $\mathfrak{D}$, i.e. select models and solve Equation (5.2) based on property estimates $\tilde{\hat{\mu}}$ instead of $\hat{\tilde{\mu}}$. However, this not only requires additional scaling calculations for every CML query, but could also result in higher estimation errors due to a lack of unification of the meta-learning output space (this will be empirically discussed in Section 5.3.3).

Naturally, one could also aim at directly meta-learning and estimating the compound scores $\hat{S}_\Omega(m, X_C)$ of all candidates. While reducing the model selector complexity, this approach also has two central drawbacks. First, changing the user priorities encoded in Ω would always require to re-evaluate the meta-learner. Second, the whole model selection would become significantly less transparent, due to not modeling each property

Table 5.1: Suggested Objectives for Evaluating CML

Ω	Description	Weights
Ω_{PCR}	balancing predictive error, complexity, and resources	$\omega_i = \begin{cases} \frac{1}{9} & \text{for MASE, RMSE, MAPE} \\ \frac{1}{6} & \text{for PAR, FS} \\ \frac{1}{12} & \text{for RTI, ENI, RTT, ENT} \end{cases}$
Ω_{QR}	balancing quality and resources	$\omega_i = \begin{cases} \frac{1}{8} & \text{for ACC, REC, PREC, F1} \\ \frac{1}{12} & \text{for PAR, FS, RTI, ENI, RTT, ENT} \end{cases}$
Ω_{MASE}	lowest error	$\omega_i = \begin{cases} 1 & \text{for MASE} \\ 0 & \text{otherwise} \end{cases}$
Ω_{ACC}	most accurate	$\omega_i = \begin{cases} 1 & \text{for ACC1} \\ 0 & \text{otherwise} \end{cases}$
Ω_{ENT}	lowest train energy	$\omega_i = \begin{cases} 1 & \text{for ENT} \\ 0 & \text{otherwise} \end{cases}$

independently from each other and thus also not being able to receive explanations as described in Section 5.1.4. The later discussed applications will also demonstrate that CML practically allows for more accurate performance estimates than Direct Meta-Learning (DML) of the compound score. For the practical evaluation of CML, it makes sense to test different user objectives Ω that correspond to specific weight assignments ω_i. Table 5.1 gives an overview for objectives used in later experiments, where respective results will be accordingly denoted for example as $\text{CML}_{\Omega_{\text{MASE}}}$ and $S_{\Omega_{\text{MASE}}}(m, C)$.

Importantly, it should be kept in mind that while CML can of course be compared to popular AutoML frameworks, it functions in very different ways. Established AutoML frameworks usually only search for solutions best on one predictive performance metric, which could for example be compared against CML with the Ω_{MASE} objective. Moreover, where AutoML approaches perform the search in an infinite model space and iteratively evaluate multiple candidates on the given data (e.g., using HPO or NAS), CML only trains the most promising models from a discrete model pool that is subject to the meta-learning database. When facing configurations (or particularly, datasets) that are very different from the ones evaluated in the database, the CML approach is thus unlikely to provide good estimates and models. That being said, CML still performed well in comparison to established AutoML systems—the following covers the respective investigations.

5.2 Application to Time Series Forecasting

Having introduced the methodological framework of CML, we now explore its practical feasibility for the first application scenario, namely selecting suitable DNNs for performing forecasting on a given time series dataset. As mentioned in Section 2.1.3, DL models

dominate the state-of-the-art for forecasting complex data, despite their obscurity and resource consumption. Moreover, the available AutoML solutions for obtaining forecasting models purely focus on predictive quality and neglect the importance of resource efficiency and transparency. To meet the need for automated and e(X)plainable selection that explicitly considers and balances model (P)rediction errors, (C)omplexity, and (R)esource consumption, our original work proposed `AutoXPCR` as the first prototype version of CML [FS24].

5.2.1 Experimental Setup

For training the meta-learners, the respective *XPCR* evaluation database (see Table 3.3) was assembled by applying $|\mathcal{P}_T| = 11$ DNNs across 108 datasets, sampled from 18 *Monash Time Series Forecasting Archive* datasets [God+21]. As only one execution environment was used, the meta-features X_C were chosen to merely describe the given dataset in terms of seasonality, frequency, and forecast horizon (as given for the original datasets), as well as some basic averaged statistics. The tested DNN models are Feed-Forward [Ale+20], DeepAR [Sal+20], N-BEATS [Ore+20], WaveNet [Oor+16], DeepState [Ran+18], DeepFactor [Wan+19], Deep Renewal Processes [Tür+21], GPForecaster [Ale+20], MQ-CNN & MQ-RNN [Wen+17], and Temporal Fusion Transformer [Lim+21]. They are all implemented via the `GluonTS` library [Ale+20] and will be respectively shortened to FFO, DAR, NBE, WVN, DST, DFA, DRP, GPF, MQC, MQR, and TFT in the following. Nine applicable properties from Table 2.1 were evaluated, however it is important to note that the original paper instead categorized the quality properties as describing the predictive errors of models. Moreover, MP and FS were not considered as describing resources, but rather represent the model complexity, with the paper arguing that automated model selection should consider and balance properties relating to predictive errors, complexity, and resource consumption (i.e., be "PCR-aware").

For meta-learning, the `XPCR` database was split into five-fold grouped cross-validation, meaning that all sampled variants of the original dataset are either used for training or validation. To make the model selector explainable by design (see Section 5.1.4), only interpretable models like linear regressors, support vector regressors, and decision trees were tested as individual meta-learners. Note that the original paper proposed two method variants, one for finding the best-balanced model (AutoXPCR) and one for finding the model with lowest prediction error (AutoXP), which will here be represented via the respective Ω_{PCR} and Ω_{MASE} objectives. In the experiments, CML was tested against a range of AutoML systems such as the forecasting extension for `AutoGluon` (AGl) [Shc+23], `AutoKeras` (AKe) [JSH19], and `Auto-Sklearn` (ASk) [Feu+22]. The complete experiment code is available in `Python`, uses `CodeCarbon` for profiling [Cou+24] and `Scikit-learn` for meta-learning [Ped+11], and can be found at `https://github.com/raphischer/xpcr`.

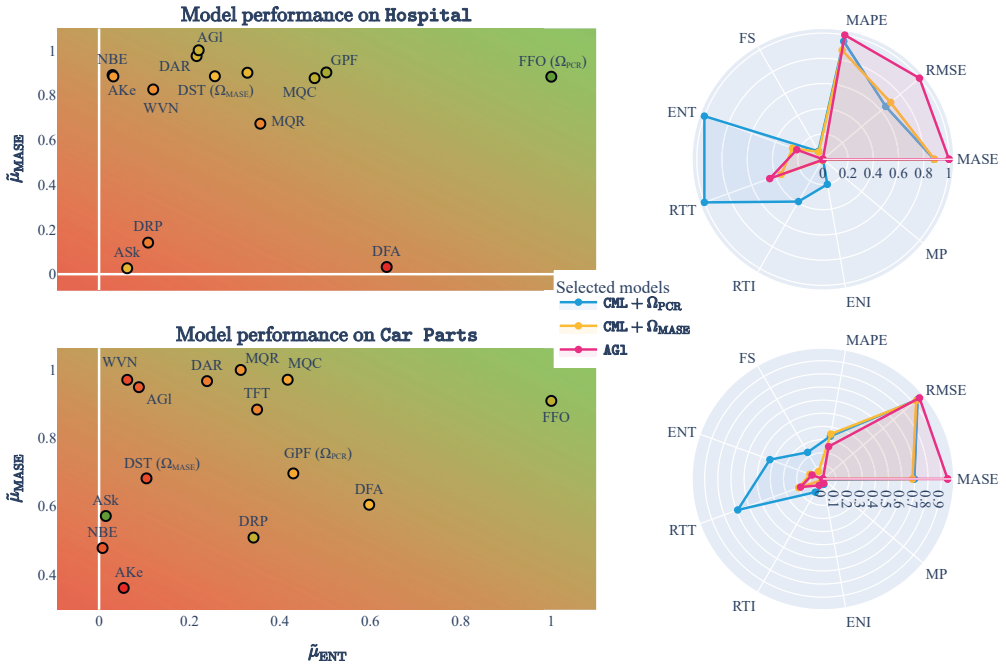

Figure 5.2: Model performance results as scatter and star plots for two selected datasets. The AutoML competitors require a lot of energy and obtain mixed predictive quality, while many DNNs achieve a better trade-off.

5.2.2 Multi-Objective Performance of DNN Forecasters

As before, we start the practical investigation by exploring how assessing model performance in multi-dimensional ways changes the understanding of state-of-the-art in forecasting. Figure 5.2 depicts an overview for the model performances for two selected datasets, clearly showing interesting trades when comparing the model training effort (ENT, x) with the error (MASE, y). Index scaling (Definition 3.2) once again unifies the properties, with higher values always indicating better performance. The models obtained from AutoML frameworks (AGl, AKe, ASk) have mixed predictive capabilities and require a lot of energy for search and training, placing them at x positions near zero. In comparison, many DNNs from the model pool seem to have a relatively good MASE while requiring much less resources, with FFO achieving especially good performance. The predicted best models obtained from CML are annotated in the scatter plot and also visualized as star plots. These traces nicely visualize that the AGl framework focuses on quality and achieves better errors rates, however the CML recommendations achieve a better general trade-off. Also recall that Figure 3.14 already featured correlation matrices for the selected datasets, representing an additional dimension for investigating the model performance across multiple properties.

Obviously, looking only at selected datasets is not enough to properly investigate the pros and cons of the evaluated models. For this reason, Figure 5.3 depicts the MASE

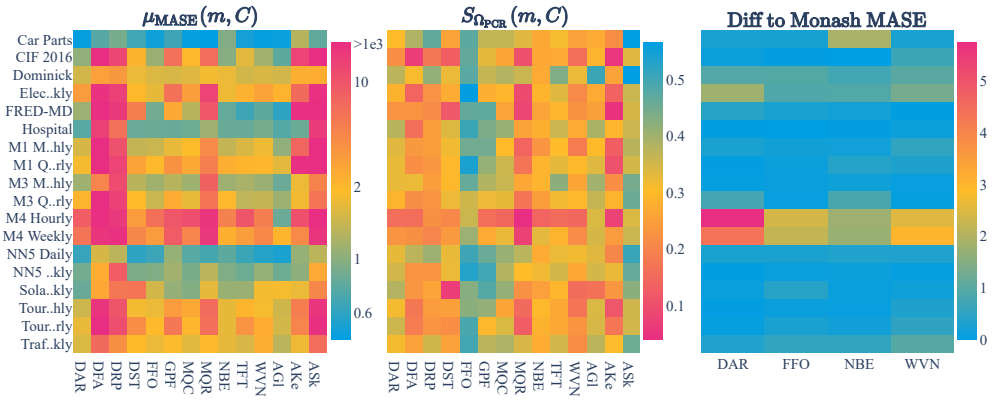

Figure 5.3: Model performance in terms of MASE (left) and compound score (middle) across datasets, with the latter dramatically changing the ranking. The right plot shows that the MASE results are close to the ones reported in the *Monash* archive [God+21].

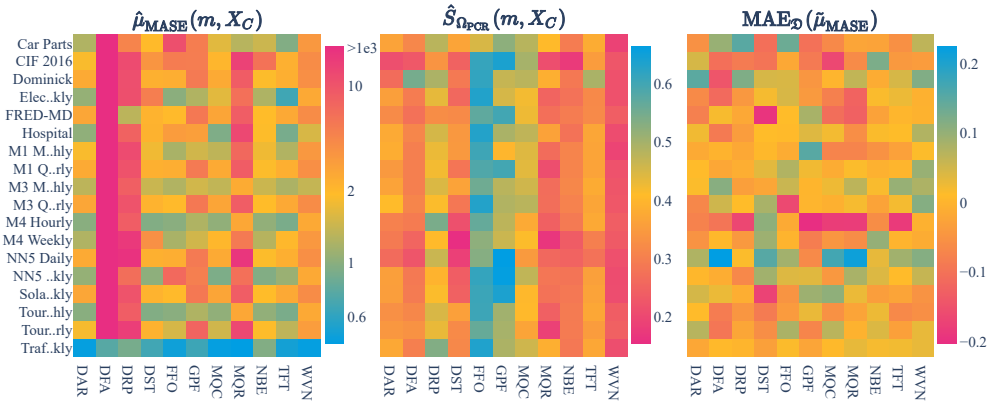

Figure 5.4: CML estimation of MASE (left) and compound score (middle), showing similar patterns as the ground-truth data and only small errors across all models and datasets (right).

and compound score (with Ω_{PCR} objective) for all models and original Monash datasets. Looking only at predictive errors (left), it is important to note that a sigmoid-like scaling was utilized to focus on the important differences among the low MASE values (which can be seen from the color bar). It reveals that for some datasets like `Car Parts` and `NN5 Daily`, nearly all models have achieved low errors, whereas they seemed to struggle to perform well for `M4 Hourly` and `M4 Weekly`. It also demonstrates that the DFA and DRP DNNs as well as models obtained by AKe and ASk have high errors across most datasets, while the AGl search provided solutions with the lowest errors.

However, looking at the model performance aggregated across all properties (middle) reveals very different patterns. The compound scoring showcases that FFO, while not performing especially well on MASE, generally archives a good performance under consideration of all properties. Moreover, the AKe competitor seems to perform better when other properties are taken into account, whereas AGl only shows a mediocre performance

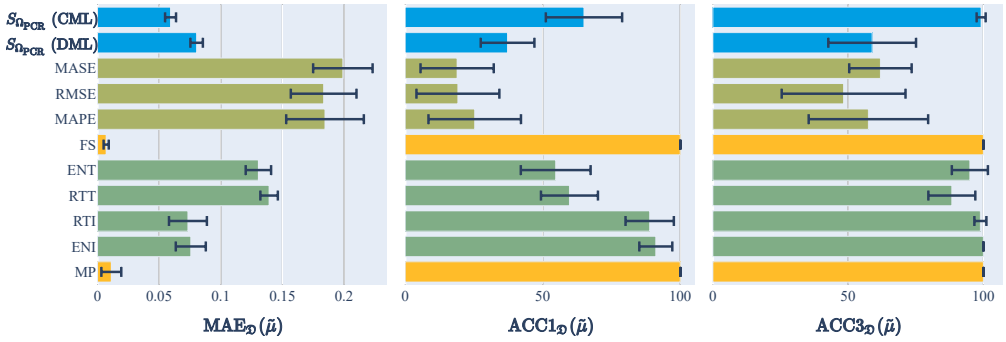

Figure 5.5: Estimations errors (left) and accuracy of correctly predicting the best candidate (middle) and having the true best model under the top-3 candidates (right). Some properties like error metrics are harder to meta-learn, resulting in higher errors and lower accuracy.

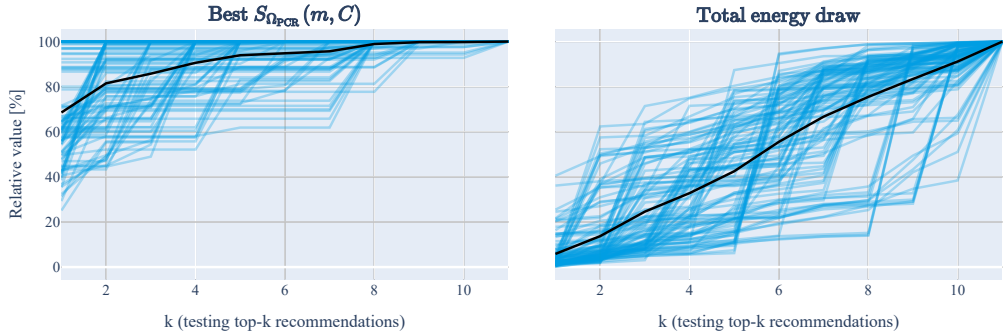

Figure 5.6: Convergence of testing the top-k recommendations received from CML. While there are deviations across the datasets (blue lines), good results (80% of the best possible compound score) can usually be found by only testing the top two candidates.

(even though it had very low errors). To make sure that our implementation and execution of the experiments aligns with the Monash benchmark [God+21], the right plot displays the MASE differences to their reported values. While they only tested four of the models that were evaluated in our study, their results generally match our results due to low differences except for some outliers. The experiments showcase that when comparing AutoML results under consideration of multiple performance aspects, the understanding of "good model performance" changes dramatically, linking back to Section 3.4 and underlining the importance of investigating resource efficiency in AutoML [Tor+23; NLA25].

5.2.3 Compositional Meta-Learning for Forecasting

For evaluating CML, the performance of all models in the DNN pool was used to meta-learn how these models are expected to perform given meta-features about the data. As a first step, once can check how well the meta-learner estimates align with the ground-truth performance. In analogy to the previous plot, Figure 5.4 therefore displays the estimated

Table 5.2: CML Performance in Comparison to Established AutoML Frameworks

Dataset	CML + Ω_{PCR}			CML + Ω_{MASE}			`AutoGluonTS`			`AutoKeras`			`AutoSklearn`		
	S_Ω	$\tilde{\mu}_{\text{MASE}}$	μ_{ENT}	S_Ω	$\tilde{\mu}_{\text{MASE}}$	μ_{ENT}	S_Ω	$\tilde{\mu}_{\text{MASE}}$	μ_{ENT}	S_Ω	$\tilde{\mu}_{\text{MASE}}$	μ_{ENT}	S_Ω	$\tilde{\mu}_{\text{MASE}}$	μ_{ENT}
Car Parts	0.37	0.70	**2.00**	0.25	0.68	8.19	0.27	**0.95**	9.76	0.22	0.36	15.7	**0.56**	0.57	57.7
CIF 2016	0.33	0.15	**0.30**	0.26	0.40	0.65	0.29	**1.00**	7.01	0.06	0.04	2.72	**0.34**	0.00	8.97
Dominick	0.47	0.95	14.4	0.37	1.00	42.2	0.50	**1.00**	**9.16**	0.25	0.67	560	**0.56**	0.66	289
Elec..kly	**0.56**	**1.00**	**0.52**	**0.56**	**1.00**	**0.52**	0.27	0.79	6.05	0.14	0.28	30.3	0.36	0.01	25.7
FRED-MD	**0.47**	0.25	**0.34**	0.12	0.14	2.51	0.25	**1.00**	6.58	0.05	0.00	8.65	0.35	0.00	10.6
Hospital	**0.50**	0.88	**1.15**	0.34	0.88	4.51	0.39	**1.00**	5.24	0.29	0.88	36.5	0.36	0.03	18.5
M1 M..hly	**0.45**	0.73	**0.89**	0.32	0.75	3.32	0.30	**1.00**	4.56	0.13	0.16	44.1	0.37	0.00	29.6
M1 Q..rly	0.41	0.52	0.41	**0.54**	0.82	**0.28**	0.34	**1.00**	3.79	0.10	0.00	4.04	0.35	0.00	9.46
M3 M..hly	**0.46**	0.75	**2.85**	0.30	0.65	15.5	0.40	**1.00**	9.33	0.22	0.49	114	0.44	0.22	83.1
M3 Q..rly	0.35	0.78	**5.68**	0.35	0.78	**5.68**	0.36	**1.00**	6.28	0.29	0.63	18.8	**0.45**	0.23	19.0
M4 Hourly	**0.36**	0.16	**0.59**	0.22	0.02	129	0.34	**1.00**	17.9	0.08	0.07	409	0.34	0.00	118
M4 Weekly	**0.41**	0.75	1.67	0.32	0.55	**1.50**	0.35	**1.00**	9.97	0.24	0.59	40.4	0.36	0.03	45.0
NN5 Daily	0.46	0.50	2.35	0.50	0.82	**0.33**	0.33	**1.00**	9.49	0.20	0.50	63.8	**0.51**	0.41	41.9
NN5 ..kly	**0.50**	0.75	**1.13**	0.35	1.00	7.67	0.35	**1.00**	5.85	0.32	0.84	5.38	0.46	0.36	38.1
Sola..kly	**0.48**	**0.79**	1.41	0.29	0.76	**0.85**	0.17	0.46	3.90	0.23	0.49	3.93	0.42	0.23	22.0
Tour..hly	**0.42**	0.85	**0.59**	0.25	0.81	5.87	0.34	**0.99**	10.5	0.11	0.34	105	0.35	0.03	25.2
Tour..rly	**0.49**	0.76	**0.32**	**0.49**	0.76	**0.32**	0.34	**1.00**	9.82	0.17	0.39	31.9	0.35	0.01	9.46
Traf..kly	**0.50**	0.77	**0.82**	0.25	0.67	71.0	0.37	**1.00**	6.51	0.30	0.75	14.6	0.45	0.23	40.4

MASE (left) and compound scores (middle) of all DNN and dataset combinations. It demonstrates that some of the discussed patterns are correctly learned, like for example the generally bad MASE of DFA and DRP, and the solid compound scores of FFO. Based on Equation (5.4), it also shows that the index-scaled error of estimating the MASE (right) seems to be rather uniformly distributed across all models and datasets.

The quality of CML can be further investigated in Figure 5.5, which features the quality metrics of Section 5.1.3 for all properties and the compound score. Highest estimation errors (left) are observed for the properties describing predictive errors and training resource consumption, which is not surprising in meta-learning. That said, the CML estimates in many cases allow to correctly guess the best candidates, which can be seen from the recommendation accuracy (middle) that for example is still near 50% for meta-learning the training resource demand. When testing the top-3 recommendations (right), the true best model can be correctly found for most properties. Finally, the plot also showcases that directly meta-learning the compound score (DML, cf. Section 5.1.5) indeed results in higher errors and worse accuracy compared to the compositional approach.

The possibility of testing multiple candidates raises the question as to how fast CML converges to the optimal solution. As shown in Figure 5.6, testing the top-2 recommendations on average results in 80% of the best possible compound score from all models in the pool, while only requiring about 15% of the energy required for performing the exhaustive search. In the plot, each blue line represents the convergence of iteratively training CML recommendations for a single investigated dataset, while the black lines represent the averages over all datasets.

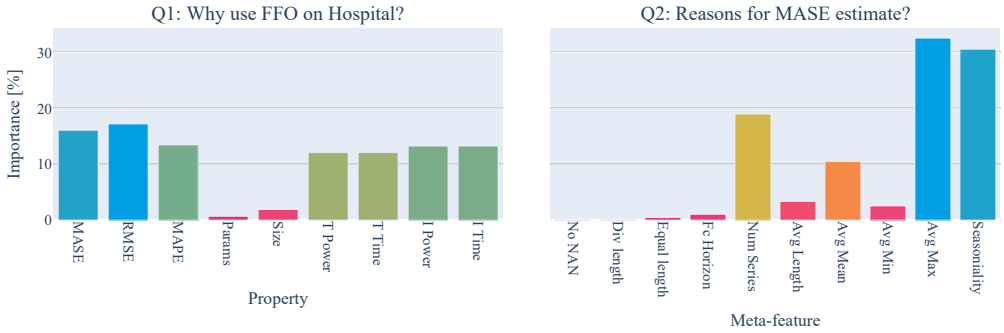

Figure 5.7: Exemplary explanations for the FFO recommendation of CML for the `Hospital` data, providing users with additional as to why a specific model should be used and which features affect the individual property estimates.

Table 5.2 provides a summarizing overview for the performance of testing the top-1 CML recommendation in comparison to models obtained from the competitor AutoML frameworks. It is striking that AGl finds models with lower errors for nearly all datasets, however the compound score (S) of these solutions is much worse than the respective score for the recommendations from the DNN pool. AKe and ASk fail to retrieve forecasters with low errors, and moreover all established AutoML methods require immense amounts of energy for the model search (i.e., training). On the contrary, CML is much more efficient, because it only trains the top recommended DNN at a mere fraction of the training costs. Moreover, it can be controlled to either focus on specific performance aspects or on a good overall balance, via adjusting the objective—the respective MASE and compound score columns demonstrate how changing the objective gives different performance results.

Finally, let us explore how the theoretical explainability of CML (see Section 5.1.4) can be put into practice. For the exemplary case of being recommended to use FFO on `Hospital` data, Figure 5.7 depicts bar plots that represent explanations for possible questions that users of CML might have. On the left, it shows which properties were most influential for the model recommendation (RMSE and MASE), and on the right, we learn which features have the strongest (global) importance for estimating the MASE of FFO.

As a conclusion to the first application study of CML [FS24], we have found strong evidence that any model selection approach for forecasting should be considerate of the diverse performance dimensions that matter in practice. Established AutoML frameworks concentrate on predictive quality and in some cases obtain models with low error rates, however their searches are very costly and moreover often return non-efficient solutions. Our findings demonstrate that training candidates from a pool of established DNNs architectures also results in high-quality forecasters that are better balanced with regard to different performance aspects. This investigation showed that the theoretical concept of CML can be translated to practice and makes the search for time series forecasters more resource-aware. On top, CML is explainable on multiple levels and allows users to infuse the search with an objective that represents their priorities.

5.3 Application to Classifying Tabular Data

For the second application scenario, we turn to the more classic problem of finding a suitable ML algorithm for a given tabular classification dataset [Fis+24]. Compared to the DNN forecasting study, this investigation has put a stronger focus on investigating differences of ML performance across multiple execution environments, based on the assumption that the choice of hardware potentially impacts the practical algorithm behavior. While also applying the CML methodology for predicting the candidates' performance on given data, additional evaluations will be conducted with regard to some other AutoML frameworks and the impact of using different meta-feature representations of the tabular datasets.

5.3.1 Experimental Setup

A key contribution of our study, also acting as a fundament for this section, is the corresponding *MetaQuRe* evaluation database [Fis+24] (see also Table 3.3). It represents the first large-scale dataset that allows to (meta-)learn how various standard ML classification algorithms perform in terms of predictive (Qu)ality and (Re)source consumption. For assembling it, a total of 8000 ML experiment configurations were evaluated, based on 200 tabular classification datasets D_T collected from various archives and domains. For all datasets, $|\mathcal{P}_T| = 10$ ML approaches were evaluated, namely kNN, SVM, RF, XRF, AB, GNB, RR, LogR, SGD, and MLP, as discussed in Section 2.1.2 and offered by `Scikit-learn` [Ped+11]. Moreover, each algorithm $\times$ dataset combination was evaluated across four execution environments (E): a 2023 workstation (*Intel* `i9-13900K` processor, 64GB RAM), a 2015 desktop (*Intel* `i7-6700`, 32GB RAM), a 2021 convertible laptop (*Intel* `i7-10610U`, 32GB RAM), and a 2023 *NVIDIA* `Jetson AGX Orin` (`ARMv8`, 64GB RAM), which is specialized for ML and AI on the edge. Note that applying the mentioned algorithms to data naturally results in usable ML models, so the earlier formalization of CML remains sound (even though the original *MetaQuRe* methodology was formalized with respect to algorithms a instead of models m [Fis+24]).

As competitors to the CML approach, the aforementioned AGl [Eri+20], as well as `Naive AutoML` (NAM) [MW23] and `TabPFN` (PFN) [Hol+23] were investigated. Recall that the latter does not represent a classic AutoML method, as it instead uses a pre-trained transformer DNN that makes predictions for the complete test set in a single forward pass, as explained in Section 2.2.3. All ML algorithms and AutoML frameworks were evaluated under consideration of ten applicable properties from Table 2.1, however the RTI and ENI here describe the resource consumption per data instance (because batch computing is not supported for all algorithms). CML was evaluated with the Ω_{ACC}, Ω_{ENT}, and Ω_{QR} objectives, as presented in Table 5.1. Moreover, four different meta-feature extraction approaches were tested, namely naive dimensionality reduction via a principal component analysis (PCA), manually constructed features inspired by [WSS16], a learned dataset representation obtained from `Dataset2Vec` (DS2VEC) [JSG21], and a combination of the latter

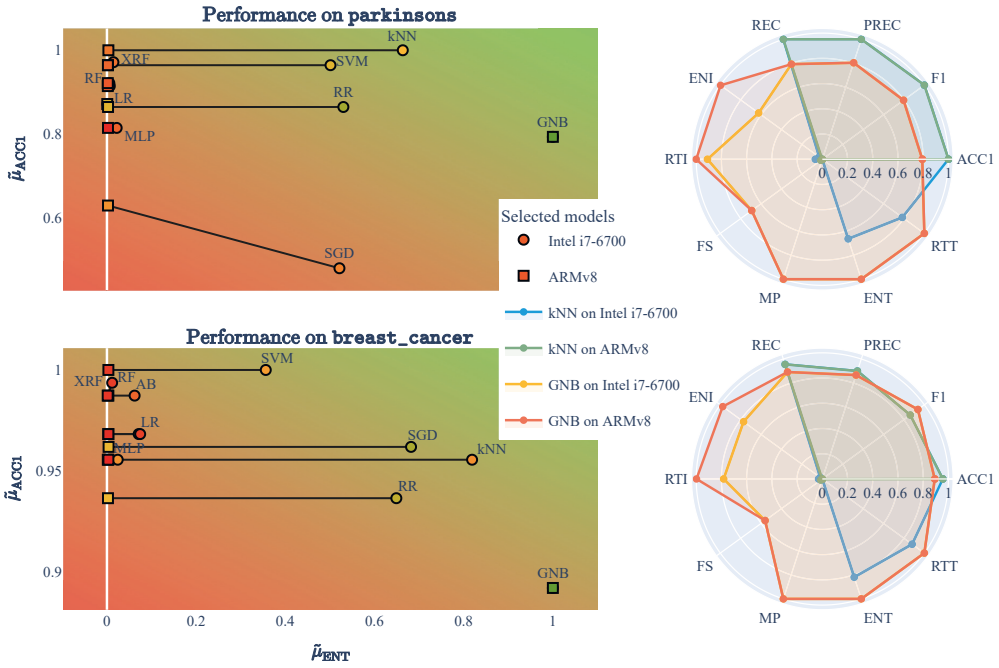

Figure 5.8: ML algorithm performance on two tabular datasets and execution environments. The relative distances between models changes drastically when being deployed on different processors, which can also be seen in the star plots for kNN and GNB.

two. In addition to these meta-features, the environment choice was one-hot encoded via four additional binary features. The remaining implementation details are comparable to the ones described Section 5.2.1, except for the specialized `NVIDIA Jetson` energy profiling that utilizes `jetson-stats`[11]. The dataset and complete code for running the experiments is available at `https://github.com/raphischer/metaqure`.

5.3.2 Insights Into MetaQuRe

We start by exploring *MetaQuRe* in similar fashion as in Section 5.2.2, however this evaluation database also allows to investigate how the choice of execution environment affects the performance of ML algorithms. Figure 5.8 displays scatter plots for two selected datasets, showcasing that the relative algorithm performance changes significantly when switching from the *Intel* processor to the `Arm` device. While GNB had the most resource-friendly training on both environments, the relative distance to the other algorithms was much larger on the `ARM`. On `parkinsons`, the kNN approach achieved the best accuracy, while SVM scores the highest quality for `breast_cancer`. Interestingly, the SGD quality is higher with `ARM` inference, which however is only observable on the `parkinsons` data. The performance differences of kNN and GNB when being deployed in different environments

[11] `https://github.com/rbonghi/jetson_stats`

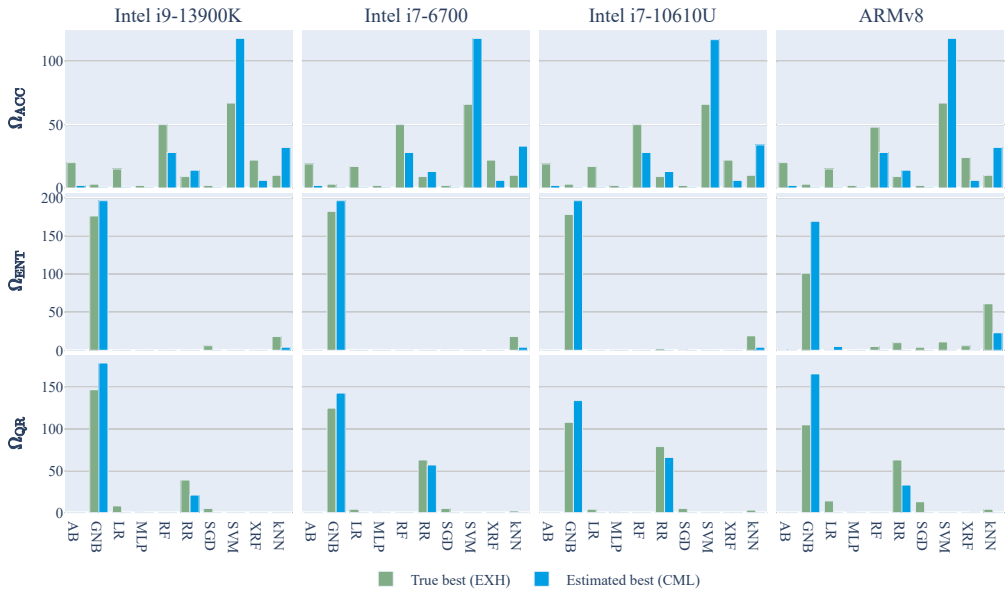

Figure 5.9: Number of datasets for which the algorithm at hand (x-axis) achieves optimal performance with regard to three search objectives (rows). Yellow bars indicate the ground-truth performance, while blue bars denote the estimated optimal models obtained from CML, which follow the true patterns in *MetaQuRe*.

can also be nicely explored from the star plots, as shown on the right. The former has a good training resource consumption on the *Intel* (70–80% of the best empiric result), however on the ARM, it performs considerably worse. GNB on the other hand behaves rather stable, with only minor differences for the inference resource consumption.

To get a better overview for the algorithm performance across all tabular datasets, Figure 5.9 depicts quantitative results for the three algorithm selection objectives. Focusing on the ground-truth information entailed in *MetaQuRe*, the green bars denote the number of datasets for which the respective algorithm (x-axis) was the optimal choice, with regard to either Ω_{ACC}, Ω_{ENT}, or Ω_{QR}. In practice, the optimal solution for any dataset and objective could be obtained by solving Equation (5.1) via an exhaustive search (EXH) over the algorithm pool. Generally, SVM and RF seem to be the most accurate candidates (first row), while GNB is extremely resource-friendly in training (second row) and also achieves a good overall balance, with RR being the second-best algorithm for balancing all performance criteria (third row). Moreover, the different columns showcase that applying algorithms in different environments does not only impact their performance, but can even lead to different optimal choices for Ω_{ENT} and Ω_{QR}. These results evidence the importance of comparing models and algorithms along multiple dimensions and also considering user priorities and deployment environments during ML engineering.

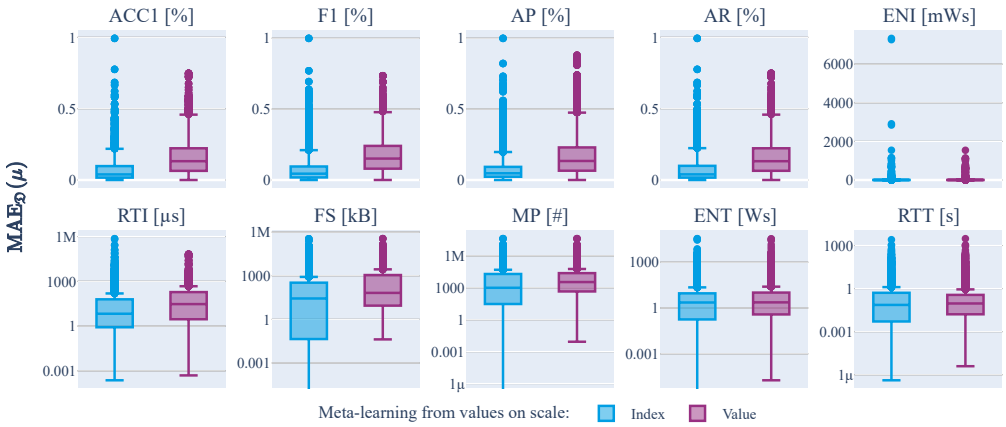

Figure 5.10: Estimation errors of all investigated properties, showcasing that meta-learning on the index scale yields lower errors than directly estimating the real-valued algorithm performance.

5.3.3 Compositional Meta-Learning From MetaQuRe

In addition to the green bars representing ground-truth performance, Figure 5.9 also gives a good overview of how CML is able to pick up the patterns of algorithm optimality (blue bars). The recommendation correctly adapts to the chosen objective and environment, however deviations from the ground-truth can also be observed. Particularly, CML seems to recommend the generally best-performing algorithms too often, indicating an over-generalization that could perhaps be improved by using additional meta-features.

To further investigate the effectiveness of CML for selecting suitable algorithms for given tabular data, Figure 5.10 displays the real-valued estimation errors introduced with Equation (5.4) for all properties. The box plots summarize the errors across all evaluations, i.e., combinations of algorithm, dataset, and execution environment. Overall, the mean errors seem to be rather low, however there are severe outliers for all properties which demonstrate the difficulty of correctly estimating algorithmic performances. As discussed in Section 5.1.5, one could also directly meta-learn the real-valued properties (purple), however compared to meta-learning the index-scaled algorithm performance (blue), higher errors can be observed across all properties.

For evaluating the impact of choosing different meta-feature extraction methods, Figure 5.11 gives an overview of the dataset representations in the form of scatter plots. Here, each point represents one specifically evaluated dataset in *MetaQuRe*. The two-dimensional coordinates were obtained via uniform manifold approximation and projection (UMAP) [HM24], showcasing that applying the meta-feature extractors results in very different embeddings. Recall that the joined meta-feature sets are merely a combination of the manual and DS2VEC meta-features, which also explains the very similar embeddings. The bar plots in the lower half demonstrate that the compound score estimation error of performing CML with different objectives is slightly impacted by the choice of

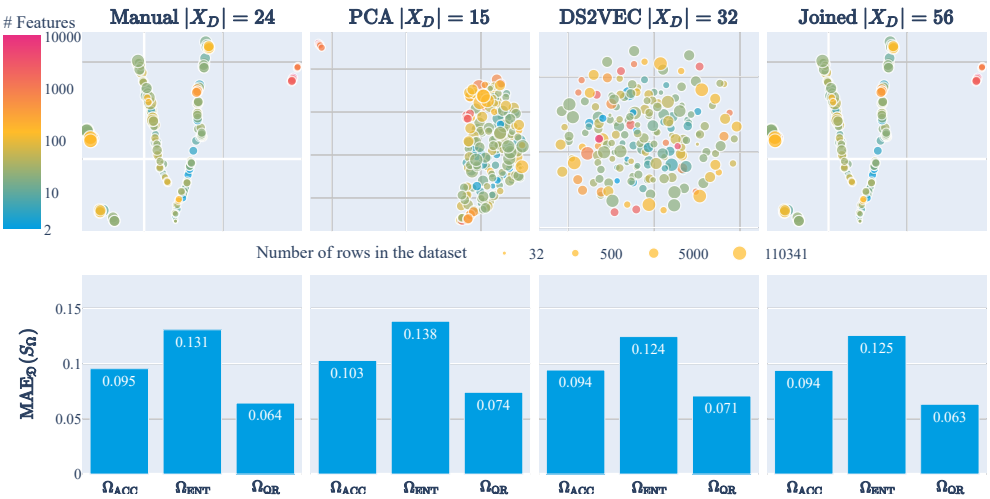

Figure 5.11: Dataset embeddings obtained from different meta-feature extractors and UMAP dimensionality reduction [HM24], along with the compound estimation errors with different search objectives. Lowest errors are archived by using a combination of the Manual and DS2VEC meta-features.

meta-features. The joined features generally obtain the lowest errors, while PCA is clearly the worst embedding.

As the last investigation, we compare how algorithm selection via CML compares to finding good models on tabular data via state-of-the-art AutoML frameworks, namely AGl [Eri+20], NAM [MW23], and PFN [Hol+23]. To enable a fair comparison with the other approaches, CML and the naive EXH were configured to select the algorithm with highest estimated quality (i.e., using the Ω_{ACC} objective). Specialized NAS approaches for obtaining DNN candidates were not evaluated because our study specifically focused on tabular data and traditional ML algorithms. In the following comparison, the *MetaQuRe* datasets were divided into small and large datasets (upper and lower half), because the original PFN model only supports datasets with limited size and complexity [Hol+23].

The performance of all approaches is visualized in Figure 5.12, in terms of the best possible accuracy and energy draw required for searching, training, and validating the candidate on the given dataset[12]. On the left, we can clearly see that the different methods achieve high and rather comparable predictive quality, with CML performing very similar to the accuracy of AGl and PFN. Interestingly, NAM retrieves the most accurate models on the more complex datasets (lower half), however also has the worst accuracy on the smaller datasets—here, the exhaustive search over our algorithm pool obtains the most

[12]Note that in the original paper, the energy draw reported in the respective figure was obtained from adding up the search energy and energy required for a *single* inference query [Fis+24]. This however puts too much focus on the training and does not adequately account for the candidate efficiency during inference, which of course is very beneficial for PFN. Looking back, it would have been fairer to sum the search efforts with the energy required to validate the model on the complete test split. For the comparison in this thesis, the figure was adjusted accordingly, resulting in differences to the original paper.

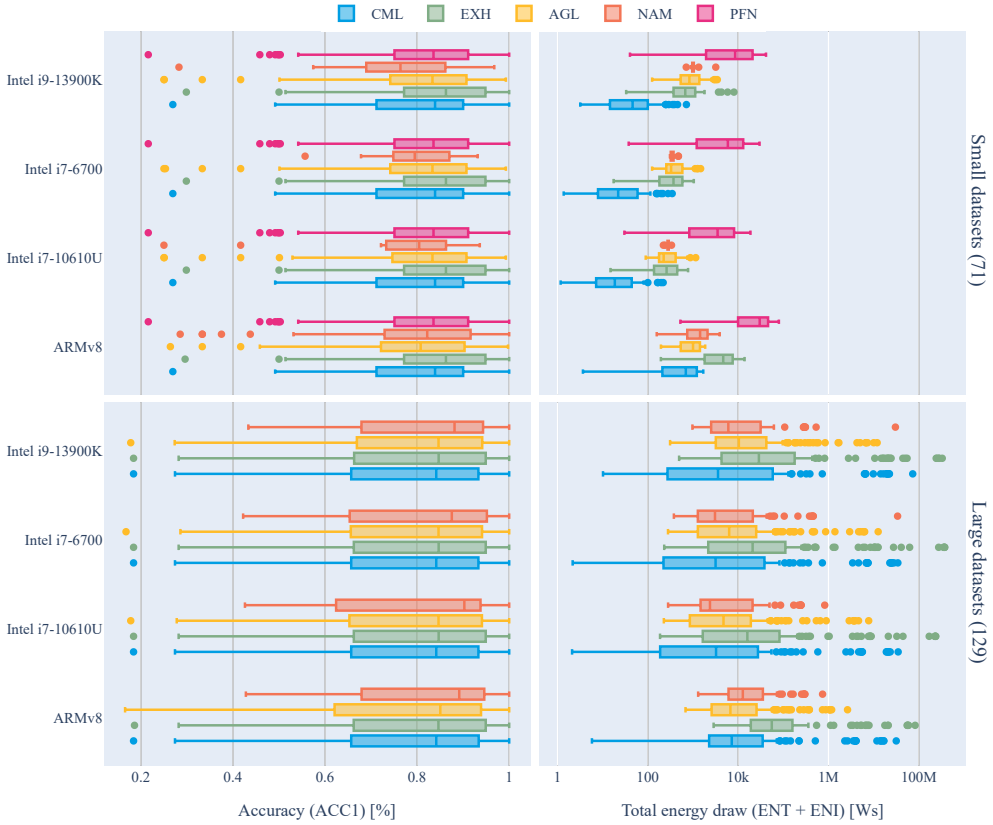

Figure 5.12: Comparison of the resulting quality and energy draw when using CML, an exhaustive search, or popular AutoML frameworks to obtain models for the *MetaQuRe* datasets. The investigated datasets had to be split because PFN only supports small datasets. On average, CML achieves high predictive quality while consuming much less resources for the search and model evaluation.

accurate models. As all approaches achieve good quality, it however is important to also compare their resource consumption, as displayed on the right (with a logarithmic x-axis). First of all, this comparison clearly demonstrates that using the CML approach is much more resource-friendly than all other approaches. The two AutoML competitors' energy consumption is comparable to EXH or even slightly more efficient for the larger datasets. We also see a clear disadvantage of PFN—due to passing the dataset through a transformer, the energy draw is immense compared to other approaches. The choice of execution environment can be seen to also affect the model selection performance, especially with regard to energy consumption. Most approaches for example require more energy on the ARM processor, however AGl seems to be slightly more efficient in this environment.

Overall, the results clearly indicate that using CML for selecting algorithms on tabular data is much more resource-efficient than using complex transformers or costly AutoML systems—our introduced method achieves comparable predictive quality while only consuming a fraction of the competitors' average resource demand. The reasonably good quality of models obtained by CML further showcases that for many tabular datasets, it is sufficient to search for basic ML algorithms, instead of building extremely complex ensembles or DNNs that are resource-heavy and non-transparent.

5.4 Conclusion

To summarize, this chapter presented and evaluated an important extension to the concept of meta-learning. Fundamentally, it allows to learn and predict model behavior for automating the search for models, aiding decision making in ML development. However, state-of-the-art AutoML methods are non-transparent and overly focus on predictive quality, which is problematic in the context of sustainability. With the formalized CML approach, meta-learning can be adapted to estimate model performance along multiple dimensions, thus balancing aspects of predictive quality and resource consumption under consideration of use case requirements. The chapter was based on the following scientific publications:

Raphael Fischer and Amal Saadallah. "AutoXPCR: Automated Multi-Objective Model Selection for Time Series Forecasting". In: *Proceedings of the 30th International Conference on Knowledge Discovery and Data Mining (KDD)*. 2024, pp. 806–815. ISBN: 979-8-4007-0490-1. DOI: 10.1145/3637528.3672057

Raphael Fischer et al. "MetaQuRe: Meta-learning from Model Quality and Resource Consumption". In: *Machine Learning and Knowledge Discovery in Databases*. 2024, pp. 209–226. ISBN: 978-3-031-70368-3. DOI: 10.1007/978-3-031-70368-3_13

The CML methodology builds on STREP concepts introduced in Chapter 3 and allows users to infuse the model selection with their own priorities and objectives. As such, they can control the model selector to either optimize specific performance aspects or achieve a general trade-off. As an additional benefit of the compositional approach,

the recommendation system becomes transparent and offers explanations to the users, enabling them to better understand the relation of data characteristics and practical ML performance. The two application scenarios demonstrated the feasibility of CML for the automated training of well-performing DNN forecasters (i.e., regression) and for choosing suitable ML algorithms to learn models for given tabular data (i.e., classification).

In comparison with various well-established AutoML frameworks, CML was shown to achieve competitive predictive quality, however behaves much more resource-efficient, interactive, and transparent. Achieving such a good practical performance in both applications does not only demonstrate the generalized capabilities of CML, but also evidences that it should indeed be considered a state-of-the-art approach for model selection. Moreover, with nearly 10000 experimental configurations, the evaluation databases *XPCR* and *MetaQuRe* represent rich information sources for future advances in AutoML and meta-learning.

Future work could apply the CML idea for automatically selecting suitable pre-trained models for given data, investigate whether preference learning [Gio+24] can be leveraged to better characterize user demands, or aim at building CML extensions for performing NAS and HPO instead of selecting models from a finite pool of candidates. The introduction and evaluation of CML represents the third central contribution of this thesis and once more addresses both issues mentioned in my general motivation. While the former chapters facilitated the sustainable use of AI, this chapter equips practitioners with means for embracing SD in the context of developing novel ML solutions.

6 Discussion

Approaching the end of my thesis, I want to once more summarize the conveyed contents and discuss them with respect to the overall thesis goals. Moreover, I will comment on the limitations of my work and describe opportunities for future extensions.

6.1 Summary of Contents

The rapid advancement of AI technology unlocks opportunities for driving positive change in our world, however also needs to be critically viewed in the context of sustainability. The transformative force of GenAI, in combination with the growing accessibility thanks to AIaaS, raises manifold questions with regard to ethical and responsible use, explainability and trustworthiness of AI models, as well as resource-awareness in the field. AI sustainability, located at the intersection of these research fields, was the focus of my PhD and this thesis, which presented novel methods and insights for advancing the development and use of AI in sustainable ways.

Goals and Contributions Two key issues were addressed for advancing sustainability, namely the imperative need for more transparency and the closely related problem of making sustainable decisions when using and developing AI. In this context, it is important to acknowledge the diversity of AI practitioners and stakeholders, requiring to explicitly bridge knowledge gaps and pay attention to various user desiderata. To establish more transparency and aid decision making, my thesis first of all conveyed a rich background on ML, AI, AutoML, and the relation to sustainability. Summarizing the related literature, Chapter 2 introduced theoretical and practical aspects of ML model development and deployment, and moreover provided an overview for the interdisciplinary discussions on risks and impacts of AI. Based on these fundamentals, the next chapters presented three central contributions for improving AI sustainability, based on eight publications from my doctorate. Chapter 3 presented `STREP`, a methodological framework for sustainable and trustworthy reporting [FLM24], accompanied by a software implementation that puts the theory into practice and allowed for practical investigations of AI efficiency and reporting biases [Fis+22; SFB24]. Chapter 4 conceptualized AI labeling as a novel approach for informing diverse stakeholders about model performance trade-offs at an easy-to-understand level [Mor+22; Mor+21], and moreover evaluated the practicability of labeling via a qualitative and interdisciplinary interview study [Mor+22]. Chapter 5 proposed CML, an approach for making meta-learning more user-centric and resource-aware, which was

also practically evaluated in two application scenarios demonstrating the potential for making AutoML more sustainable [FS24; Fis+24].

Theoretical Considerations The three central chapters of my thesis made several important theoretical contributions in the form of formalizations and algorithms, which are interconnected along several lines and together advance AI sustainability. The STREP methodology for the characterization, index scaling, compound scoring, categorical rating, and correlation analysis of model properties can establish a novel and unified understanding of AI performance, and thus, sustainable reporting. Importantly, this methodological framework is agnostic to learning tasks or data domains and thus holistically benefits AI sustainability.

The STREP methods are also of central performance for practically implementing the fundamental idea of AI labeling, because it ultimately allows to communicate intricate technical information like quality metrics and resource consumption via color-coded icons. The thematic analysis of practitioner positions evaluated the theoretical concept of AI labels and underscored the importance of comprehensibility and customizability, directly linking to the respective STREP steps.

For model selection, the STREP formalizations are key for scoring and (meta-)learning model suitability in multi-objective ways and enable the proposed CML approach. It not only allows for making an educated guess regarding which model should be actually tested on the given data, but also empowers users to infuse the model selection with their own objectives and thoroughly understand why a specific model was recommended.

Importantly, the STREP formalization for analyzing property correlation and evaluation biases can also be applied in the context of labeling and meta-learning, where it provides additional information on the quality and contents of the evaluation (or meta-learning) database at hand. As such, the thesis contributed a strong set of formalized methods for making the development and use of ML and AI more resource-aware, transparent, customizable, and thus, sustainable.

Practical Insights In addition to the proposed theory, the thesis was written with the explicit goal of providing practical benefits, with the STREP software arguably being the biggest contribution. It makes the respective methodology available for practical use and offers to interactively explore several pre-compiled evaluation databases, containing model performance results that are considerate of resource efficiency trade-offs. Moreover, users can also easily investigate their own custom databases and compare the model performance along the respectively evaluated properties. STREP also automatically generates labels for given evaluation database entries and was the starting point for putting CML into practice.

In terms of experimental results, Section 3.4.1 started with investigating the resource efficiency of pre-trained `ImageNet` models. The results demonstrated the feasibility of STREP

for unifying various practical performance aspects, with `MobileNet` variants seemingly being the most sustainable option, due to achieving a good balance between predictive quality and resource consumption. The next application (Section 3.4.2) investigated the resource efficiency of USB accelerator processors, which are marketed as universal solutions for efficient edge inference and indeed provided resource improvements for many models. However, the experiments also unveiled occasional negative effects, guiding practitioners with helpful insights for accelerated edge inference and evidencing that the NFL theorem should also be acknowledged in the context of execution environments. Section 3.4.3 moved away from individual model performances and instead investigated property correlations and biases. While the custom assembled `STREP` databases also describe performance trade-offs, the public databases were shown to be ill-balanced. Mostly reporting on strongly correlated and redundant performance aspects, this evidences the need for more sustainable reporting.

Chapter 4 formulated practical insights based on the labeling interview study, which overall found labels to be an important communication format for connecting AI stakeholders. As an important takeaway, labeling approaches should focus on (1) balancing the simplicity versus complexity trade-off, (2) acknowledging the suitability of labels for nudging practitioners toward SD, (3) deeper investigating their role for trust-building processes, and (4) establishing means for customization, under consideration of various stakeholder desiderata. These findings can be used to derive decision principles for advancing and refining the concept of labeling AI models. In Chapter 5, the experiments successfully demonstrated the practical feasibility of CML in two application domains. For forecasting time series with DNNs, the rather basic feed-forward model was shown to be most resource-efficient and CML proved to be competitive with established AutoML approaches, making sophisticated model recommendations with respect to the meta-features and objective at hand. This was reinforced in the second application, which investigated the pros and cons of basic ML algorithms for training classifiers on tabular data. The results showcased how algorithm suitability ultimately depends on the objective and environment at hand, and also found CML to be resource-efficient and adaptable to that end, delivering state-of-the-art predictive performance while being much less computationally demanding than popular AutoML frameworks.

6.2 Limitations and Future Work

Naturally, researchers and scientists know the flaws and shortcomings of their works best, and in order to be self-critical and transparent, I would like to discuss some limitations of my thesis and explain the resulting opportunities for future work.

For a start, my thesis purposefully focused on practical AI model properties, however theoretical considerations can also be of relevance. As explained in Section 4.1.1, our conceptual care label work therefore proposed to also analyze and communicate the theoretic properties of various ML methods. The immense complexity of the ML field however

makes it hard to assemble such expert knowledge, requiring a larger research group or community that collects and characterizes theoretic bounds and guarantees. Moreover, theory does not necessarily translate to the code implementation, so in-depth testing would be required to reliably inform practitioners looking for practical guidelines based on theory. In a way, my thesis represents a top-down approach for discussing ML performance, focusing on practically measurable properties without an explicit consideration of underlying theory. Thoroughly analyzing the fundamentals for building corresponding expert knowledge and practical software tests would be the corresponding bottom-up approach, which could be followed in future work.

As a second limitation, my thesis introduction and background acknowledged all dimensions of sustainability, however the practical investigations put a strong focus on energy efficiency, and thus, environmental aspects. It should be noted that others have argued this dimension to be the most pressing one to investigate in the context of AI sustainability [Wyn21], however assessing, comparing, and communicating model properties with regard to implications for society and economy would be an important extension to my work. Thanks to the holistic nature of STREP, these factors and trade-offs could be easily investigated once respective model properties are quantified. Future work could therefore identify or develop specialized numerical measures or metrics that for example describe the trustworthiness, explainability, fairness, or safety of specific ML models.

As a rather technical limitation of STREP, the index scaling as proposed in Section 3.2.2 maps real-valued properties onto the unit scale with linear dependency. However, Equation (3.2) could be extended to also perform non-linear projections whenever the observed properties follow respective patterns, like logarithmic scaling. This might further improve the unification of properties and could allow for a better comparison of model performance. Moreover, in the context of performing meta-learning on the properties, index scaling with different scales could be understood as a customizable preprocessing of the regression target variable, which might result in more accurate meta-learning results.

The recommended labeling practices formulated in Section 4.2.3 represent an obvious opportunity for future work, which has partly been addressed in our already published follow-up work on reflective design theorizing [Sta+25]. To be more specific, one could for example use grouped compound scoring (Definition 3.4) to make the labels less technical and only display the relative performance along abstract property groups. Our qualitative study provided interesting insights, however a quantitative follow-up analysis could be beneficial for solidifying the derived theories. Learning from other labeling systems and going into interdisciplinary exchange with regulatory authorities and policymakers is necessary for making AI labels a trustworthy and well-accepted communication format.

Besides the label generation, various other extensions to the STREP software are possible, such as constructing additional plots or adding new evaluation databases, for example based on *Hugging Face* models and leaderboards. Additional user studies could be performed to evaluate and improve the current software and exploration tool. With STREP as the fundament, a unified AutoML tool could be implemented that uses the CML approach to recommend models for user-provided datasets. Moreover, the idea behind CML could

be extended toward other tasks than selecting models from a finite set of candidates, for example addressing HPO or NAS.

Finally, I want to also shortly comment on the impact of my work in the global context of AI evolution, which was a re-occurring point of discussion during my PhD. I consider my research work to be of fundamental nature, and as such, it might not necessarily have a big impact in the publicly observable AI market and hype. That said, my papers and the labeling idea in particular were positively received by the research community, sparked adaptions, and could also be easily leveraged to also make AIaaS more sustainable. To that end, I personally see the corporate power of AI service providers as an immense problem—they alone have the data, finances, and expertise to shape the future of AI, and in my humble opinion, they do not sufficiently consider the importance of sustainability. Connecting back to the Jevons' paradox discussed in Section 2.3.2, ML research can aim at establishing resource-awareness and developing more efficient models, however will likely not stop the observable compute trends. I therefore believe that global policymaking is needed, establishing clear guidelines that incentivize responsible innovation while penalizing harmful practices such as unchecked data exploitation, extreme compute expenses, or opaque algorithmic decision-making processes. Simultaneously, educational institutions need to emphasize interdisciplinary training and bridge gaps between AI engineers and experts from other domains, like ethicists, sociologists, and economists—ensuring tomorrow's AI developers are equipped with both technical skills and moral foresight.

To summarize, AI sustainability is still far from being generally established, and as with any research, my own work also faces limitations. However, a lot of these issues directly open opportunities for future work on making AI more sustainable. In this context, I am happy to have recently joined the global *AI for Good* network as a *Young AI Leader*[13], building a local community that empowers young minds to drive positive change with AI. Maybe this network also finds inspiration in my thesis, and generally I look forward to further collaborations on the discussed topics.

6.3 Closing Words

In conclusion, my thesis is motivated by the dual nature of modern AI—while arguably having the power to make our world a better place, AI technology needs to be used in ethical and responsible ways. From environmental costs associated with large-scale models to societal concerns like bias and inequity, it is evident that advancing AI sustainability requires much more than small technical improvements. With the growing model complexity and the new paradigm of AIaaS in particular, establishing transparency is key for making informed and sustainable decisions in AI development and deployment.

My thesis addressed these issues via contributions for advancing AI sustainability, particularly formalizing a sound methodology for reporting, introducing comprehensible AI

[13] https://youngaileadersdortmund.github.io/

labels, and proposing a novel take on AutoML with the help of meta-learning. In combination, these approaches make communication and decision making more user-centric, resource-aware, and trustworthy. While also presenting theoretic formalizations and concepts, my work primarily provided practical insights for making the development and use of AI more sustainable.

Importantly, sustainability in our field is not at an end point, but must be understood as an evolving journey that requires continuous reflection and adaptation. Researchers like myself can make AI more efficient, develop means for establishing more transparency, or propose practices for SD—however, I believe that real change necessitates a commitment on a global political and economic level, which likely can only be achieved through the establishment of policies and regulations. Nevertheless, looking back on my PhD and my work as a member of the research community, I am extremely proud to have contributed to the cause of AI sustainability. I am confident that my research publications, and this thesis in particular, are helpful resources for solidifying the spirit of SD in the field, and hope that they will make future AI progress more sustainable.

List of Figures

List of Tables

List of Algorithms

Bibliography

[AB18] Amina Adadi and Mohammed Berrada. "Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)". In: *IEEE Access* (2018), pp. 52138–52160. DOI: 10.1109/ACCESS.2018.2870052.

[Aba+16] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*. 2016. URL: https://arxiv.org/abs/1603.04467.

[ADW17] Zahraa S. Abdallah, Lan Du, and Geoffrey I. Webb. "Data Preparation". In: *Encyclopedia of Machine Learning and Data Mining*. 2017, pp. 318–327. ISBN: 978-1-4899-7687-1. DOI: 10.1007/978-1-4899-7687-1_62.

[Alc05] Blake Alcott. "Jevons' Paradox". In: *Ecological Economics* (July 2005), pp. 9–21. ISSN: 0921-8009. DOI: 10.1016/j.ecolecon.2005.03.020.

[Ale+20] Alexander Alexandrov et al. "GluonTS: Probabilistic and Neural Time Series Modeling in Python". In: *Journal of Machine Learning Research* (2020), pp. 1–6. URL: https://jmlr.org/papers/v21/19-820.html.

[Ali+24] Mehdi Ali et al. *Teuken-7B-Base & Teuken-7B-Instruct: Towards European LLMs*. 2024. URL: https://arxiv.org/abs/2410.03730.

[Als+22] Ahmad Alsharef et al. "Review of ML and AutoML Solutions to Forecast Time-Series Data". In: *Archives of Computational Methods in Engineering* (Nov. 2022), pp. 5297–5311. ISSN: 1886-1784. DOI: 10.1007/s11831-022-09765-0.

[Alt+24] Patrick Altmeyer et al. "Position: Stop Making Unscientific AGI Performance Claims". In: *Proceedings of the 41st International Conference on Machine Learning (ICML)*. July 2024, pp. 1222–1242. URL: https://proceedings.mlr.press/v235/altmeyer24a.html.

[Ang+21] Plamen P. Angelov et al. "Explainable Artificial Intelligence: An Analytical Review". In: *WIREs Data Mining Knowl. Discov.* (2021). DOI: 10.1002/WIDM.1424.

[Arn+19] Matthew Arnold et al. "FactSheets: Increasing Trust in AI Services Through Supplier's Declarations of Conformity". In: *IBM Journal of Research and Development* (Sept. 2019), 6:1–6:13. ISSN: 0018-8646. DOI: 10.1147/JRD.2019.2942288.

[Arn+22] Matthew R. Arnold et al. "Generation and Management of Artificial Intelligence (AI) Model Documentation Throughout Its Lifecycle". Mar. 2022. URL: https://patents.google.com/patent/US11263188B2/en.

[Arn+24] Stefan Arnold et al. *Documentation Practices of Artificial Intelligence*. 2024. URL: https://arxiv.org/abs/2406.18620.

[Art24] Artifex Software, Inc. *PyMuPDF: Python Bindings for MuPDF*. Accessed: 2025-08-13. 2024. URL: https://pymupdf.readthedocs.io/.

[Avi+21] Shahar Avin et al. "Filling Gaps in Trustworthy Development of AI". In: *Science* (Dec. 2021), pp. 1327–1329. DOI: 10.1126/science.abi7176.

[Bak+17] Bowen Baker et al. "Designing Neural Network Architectures using Reinforcement Learning". In: *5th International Conference on Learning Representations (ICLR)*. 2017. URL: https://openreview.net/forum?id=S1c2cvqee.

[Bal+23] Randall Balestriero et al. *A Cookbook of Self-Supervised Learning*. 2023. URL: https://arxiv.org/abs/2304.12210.

[Bar+20] Alejandro Barredo Arrieta et al. "Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges Toward Responsible AI". In: *Information Fusion* (June 2020), pp. 82–115. ISSN: 1566-2535. DOI: 10.1016/j.inffus.2019.12.012.

[BB12] James Bergstra and Yoshua Bengio. "Random Search for Hyper-Parameter Optimization". In: *Journal of Machine Learning Research* (2012), pp. 281–305. URL: https://jmlr.org/papers/v13/bergstra12a.html.

[BB19] Shikha Bordia and Samuel R. Bowman. "Identifying and Reducing Gender Bias in Word-Level Language Models". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*. June 2019, pp. 7–15. DOI: 10.18653/v1/N19-3002.

[Bec+23] Katharina Beckh et al. "Harnessing Prior Knowledge for Explainable Machine Learning: An Overview". In: *1st IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. 2023.

[Ben+21] Hadjer Benmeziane et al. "Hardware-Aware Neural Architecture Search: Survey and Taxonomy". In: *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*. Survey Track. Aug. 2021, pp. 4322–4329. DOI: 10.24963/ijcai.2021/592.

[Ben+24] Yoshua Bengio et al. "Managing Extreme AI Risks Amid Rapid Progress". In: *Science* (May 2024), pp. 842–845. DOI: 10.1126/science.adn0117.

[Ben+25] Yoshua Bengio et al. *International AI Safety Report*. Tech. rep. 2025. URL: https://coilink.org/20.500.12592/30e27x2.

[Bid23] Joseph R. Biden. *Executive order on the safe, secure, and trustworthy development and use of artificial intelligence*. Accessed: 2025-08-13. 2023. URL: https://www.federalregister.gov/d/2023-24283/.

[Bie23] Celeste Biever. "ChatGPT broke the Turing test-the race is on for new ways to assess AI". In: *Nature* (2023), pp. 686–689. DOI: 10.1038/d41586-023-02361-7.

[Bir+22] Abeba Birhane et al. “The Values Encoded in Machine Learning Research”. In: *Proceedings of the 5th Conference on Fairness, Accountability and Transparency (FAccT)*. 2022, pp. 173–184. ISBN: 978-1-4503-9352-2. DOI: 10.1145/3531146.3533083.

[Boa+18] Scott Boag et al. “Dependability in a Multi-tenant Multi-framework Deep Learning as-a-Service Platform”. In: *2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*. 2018, pp. 43–46. DOI: 10.1109/DSN-W.2018.00022.

[Bom+22] Rishi Bommasani et al. *On the Opportunities and Risks of Foundation Models*. 2022. URL: https://arxiv.org/abs/2108.07258.

[Bre+23] Kathrin Brecker et al. “Artificial Intelligence as a Service: Trade-Offs Impacting Service Design and Selection”. In: *ICIS 2023 Proceedings. 5, Hyderabad, 10th-13th December 2023*. 2023, p. 17.

[Bro+20] Tom Brown et al. “Language Models are Few-Shot Learners”. In: *Proceedings of the 34th International Conference on Neural Information Processing Systems (NeurIPS)*. 2020, pp. 1877–1901. URL: https://dl.acm.org/doi/abs/10.5555/3495724.3495883.

[Bro06] David C. Brock. *Understanding Moore’s Law: Four Decades of Innovation*. Chemical Heritage Foundation, 2006. ISBN: 978-0-941901-41-3. URL: https://books.google.de/books?id=woBkE-_SOCUC.

[Bru+20] Miles Brundage et al. *Toward Trustworthy AI Development: Mechanisms for Supporting Verifiable Claims*. 2020. URL: https://arxiv.org/abs/2004.07213.

[BS93] Thomas Bäck and Hans-Paul Schwefel. “An Overview of Evolutionary Algorithms for Parameter Optimization”. In: *Evolutionary Computation* (1993), pp. 1–23. ISSN: 1063-6560. DOI: 10.1162/evco.1993.1.1.1.

[Bud+22] Semen Budennyy et al. “eco2AI: Carbon Emissions Tracking of Machine Learning Models as the First Step Towards Sustainable AI”. In: *Doklady Mathematics* (Dec. 2022). ISSN: 1531-8362. DOI: 10.1134/S1064562422060230.

[Bus+18] Sebastian Buschjager et al. “Realization of Random Forest for Real-Time Evaluation through Tree Framing”. In: *2018 IEEE International Conference on Data Mining (ICDM)*. 2018, pp. 19–28. DOI: 10.1109/ICDM.2018.00017.

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge university press, 2004. ISBN: 9780521833783. URL: https://cambridge.org/9780521833783.

[BX23] Stephanie Baker and Wei Xiang. *Explainable AI is Responsible AI: How Explainability Creates Trustworthy and Socially Responsible Artificial Intelligence*. 2023. URL: https://arxiv.org/abs/2312.01555.

[Cas+23] Joel Castaño et al. “Exploring the Carbon Footprint of Hugging Face’s ML Models: A Repository Mining Study”. In: *2023 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*. Oct. 2023, pp. 1–12. DOI: 10.1109/ESEM56168.2023.10304801.

[CB23] Tara Capel and Margot Brereton. “What is Human-Centered about Human-Centered AI? A Map of the Research Landscape”. In: *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 2023. ISBN: 978-1-4503-9421-5. DOI: 10.1145/3544548.3580959.

[Cha+21] Raja Chatila et al. “Trustworthy AI”. In: *Reflections on Artificial Intelligence for Humanity* (2021), pp. 13–39. DOI: 10.1007/978-3-030-69128-8_2.

[Cha24] Eya Ben Chaaben. “Exploring Human-AI Collaboration and Explainability for Sustainable ML”. In: *HHAI 2024: Hybrid Human AI Systems for the Social Good - Proceedings of the Third International Conference on Hybrid Human-Artificial Intelligence, Malmö, Sweden, 10-14 June 2024*. 2024, pp. 363–370. DOI: 10.3233/FAIA240209.

[Chm+22] Kasia S. Chmielinski et al. *The Dataset Nutrition Label (2nd Gen): Leveraging Context to Mitigate Harms in Artificial Intelligence*. 2022. URL: https://arxiv.org/abs/2201.03954.

[Cho+20] Tejalal Choudhary et al. “A Comprehensive Survey on Model Compression and Acceleration”. In: *Artificial Intelligence Review* (Oct. 2020), pp. 5113–5155. ISSN: 1573-7462. DOI: 10.1007/s10462-020-09816-7.

[Cho+25] Francois Chollet et al. *ARC Prize 2024: Technical Report*. 2025. URL: https://arxiv.org/abs/2412.04604.

[Cho17] François Chollet. “Xception: Deep Learning with Depthwise Separable Convolutions”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 1800–1807. DOI: 10.1109/CVPR.2017.195.

[Cho19] François Chollet. *On the Measure of Intelligence*. 2019. URL: https://arxiv.org/abs/1911.01547.

[Col15] William Colglazier. “Sustainable Development Agenda: 2030”. In: *Science* (2015), pp. 1048–1050. DOI: 10.1126/science.aad2333.

[Cor24] Coral. *Coral USB Accelerator: High-Speed Machine Learning Inferencing via USB*. Accessed: 2025-08-13. 2024. URL: https://coral.ai/products/accelerator/.

[Cou+24] Benoit Courty et al. *mlco2/codecarbon: v2.4.1*. May 2024. DOI: 10.5281/zenodo.11171501.

[Cre+19] Armin B. Cremers et al. “Trustworthy Use of Artificial Intelligence-Priorities From a Philosophical, Ethical, Legal, and Technological Viewpoint as a Basis For Certification of Artificial Intelligence”. In: *Fraunhofer Institute for Intelligent Analysis and Information Systems (IAIS) Whitepapers* (2019). DOI: 10.24406/publica-184.

[Cro+21] Francesco Croce et al. "RobustBench: A Standardized Adversarial Robustness Benchmark". In: *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*. 2021. URL: https://openreview.net/forum?id=SSKZPJCt7B.

[Cro24] Caroline Crosdale. *Microsoft Taps Nuclear Power To Fuel Growing AI Demand*. Accessed: 2025-08-13. Oct. 2024. URL: https://gfmag.com/economics-policy-regulation/microsoft-three-mile-island-nuclear-power-ai-demand/.

[Cui19] Wenqiang Cui. "Visual Analytics: A Comprehensive Overview". In: *IEEE Access* (2019), pp. 81555–81573. DOI: 10.1109/ACCESS.2019.2923736.

[Dem06] Janez Demšar. "Statistical Comparisons of Classifiers over Multiple Data Sets". In: *Journal of Machine Learning Research* (2006), pp. 1–30. URL: https://jmlr.org/papers/v7/demsar06a.html.

[Den+09] Jia Deng et al. "Imagenet: A Large-Scale Hierarchical Image Database". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[Di +20] Assunta Di Vaio et al. "Artificial Intelligence and Business Models in the Sustainable Development Goals Perspective: A Systematic Literature Review". In: *Journal of Business Research* (Dec. 2020), pp. 283–314. ISSN: 0148-2963. DOI: 10.1016/j.jbusres.2020.08.019.

[Dig19] Virginia Dignum. *Responsible Artificial Intelligence: How to Develop and Use AI in a Responsible Way*. Springer, 2019. URL: https://link.springer.com/book/10.1007/978-3-030-30371-6.

[DR20] Rébecca Deneckère and Gregoria Rubio. "EcoSoft: Proposition of an Eco-Label for Software Sustainability". In: *Advanced Information Systems Engineering Workshops*. 2020, pp. 121–132. ISBN: 978-3-030-49165-9. DOI: 10.1007/978-3-030-49165-9_11.

[DSH15] Tobias Domhan, Jost Tobias Springenberg, and Frank Hutter. "Speeding up Automatic Hyperparameter Optimization of Deep Neural Networks by Extrapolation of Learning Curves". In: *Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI)*. 2015, pp. 3460–3468. ISBN: 9781577357384. URL: https://dl.acm.org/doi/10.5555/2832581.2832731.

[Dur+24] Pau Duran et al. "GAISSALabel: A Tool for Energy Labeling of ML Models". In: *Companion Proceedings of the 32nd ACM International Conference on the Foundations of Software Engineering*. 2024, pp. 622–626. ISBN: 979-8-4007-0658-5. DOI: 10.1145/3663529.3663811.

[EA24] United Nations Department of Economic and Social Affairs. *The Sustainable Development Goals Report 2024*. 2024th ed. United Nations, 2024. URL: https://un-ilibrary.org/content/books/9789213589755.

[EMH19a] Thomas Elsken, Jan Hendrik Metzen, and Frank Hutter. "Neural Architecture Search". In: *Automated Machine Learning: Methods, Systems, Challenges*. 2019, pp. 63–77. ISBN: 978-3-030-05318-5. DOI: 10.1007/978-3-030-05318-5_3.

[EMH19b] Thomas Elsken, Jan Hendrik Metzen, and Frank Hutter. "Neural Architecture Search: A Survey". In: *Journal of Machine Learning Research* (Jan. 2019), pp. 1997–2017. ISSN: 1532-4435. URL: http://jmlr.org/papers/v20/18-598.html.

[EMS09] Hugo Jair Escalante, Manuel Montes, and Luis Enrique Sucar. "Particle Swarm Model Selection". In: *Journal of Machine Learning Research* (2009), pp. 405–440. URL: https://jmlr.org/papers/v10/escalante09a.html.

[Eri+20] Nick Erickson et al. *AutoGluon-Tabular: Robust and Accurate AutoML for Structured Data*. 2020. URL: https://arxiv.org/abs/2003.06505.

[ES20] Peter Elger and Eóin Shanaghy. *AI as a Service: Serverless machine learning with AWS*. Manning, 2020. ISBN: 978-1-61729-615-4. URL: https://books.google.de/books?id=BeT7DwAAQBAJ.

[Eur23] European Parliament. *A step closer to the first rules on Artificial Intelligence*. European Parliament News. 2023. URL: https://europarl.europa.eu/news/en/press-room/20230505IPR84904/ai-act-a-step-closer-to-the-first-rules-on-artificial-intelligence.

[Eur24a] European Commission. *Understanding the Energy Label*. Accessed: 2025-08-13. 2024. URL: https://energy-efficient-products.ec.europa.eu/ecodesign-and-energy-label/understanding-energy-label_en.

[Eur24b] European Union Directorate General for Internal Market, Industry, Entrepreneurship and SMEs. *Textile Label*. Accessed: 2025-08-13. 2024. URL: https://europa.eu/youreurope/business/product-requirements/labels-markings/textile-label.

[Feu+15] Matthias Feurer et al. "Efficient and Robust Automated Machine Learning". In: *Proceedings of the 29th International Conference on Neural Information Processing Systems (NeurIPS)*. 2015. URL: https://dl.acm.org/doi/10.5555/2969442.2969547.

[Feu+22] Matthias Feurer et al. "Auto-Sklearn 2.0: Hands-free AutoML via Meta-Learning". In: *Journal of Machine Learning Research* (2022), pp. 1–61. URL: https://jmlr.org/papers/v23/21-0992.html.

[Feu+24] Stefan Feuerriegel et al. "Generative AI". In: *Business & Information Systems Engineering* (Feb. 2024), pp. 111–126. ISSN: 1867-0202. DOI: 10.1007/s12599-023-00834-7.

[FH19] Matthias Feurer and Frank Hutter. "Hyperparameter Optimization". In: *Automated Machine Learning: Methods, Systems, Challenges*. 2019, pp. 3–33. ISBN: 978-3-030-05318-5. DOI: 10.1007/978-3-030-05318-5_1.

[Fis+20a] Raphael Fischer et al. “No Cloud on the Horizon: Probabilistic Gap Filling in Satellite Image Series”. In: *IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*. 2020, pp. 546–555. DOI: 10.1109/DSAA49011.2020.00069.

[Fis+20b] Raphael Fischer et al. “Solving Abstract Reasoning Tasks with Grammatical Evolution”. In: *LWDA Workshops: FGWM, KDML, FGWI-BIA, and FGIR*. Sept. 2020, pp. 6–10. URL: https://ceur-ws.org/Vol-2738/LWDA2020_paper_8.pdf.

[Fis+22] Raphael Fischer et al. “A Unified Framework for Assessing Energy Efficiency of Machine Learning”. In: *Machine Learning and Principles and Practice of Knowledge Discovery in Databases*. 2022, pp. 39–54. DOI: 10.1007/978-3-031-23618-1_3.

[Fis+23] Raphael Fischer et al. “Prioritization of Identified Data Science Use Cases in Industrial Manufacturing via C-EDIF Scoring”. In: *2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA)*. 2023, pp. 1–4. DOI: 10.1109/DSAA60987.2023.10302632.

[Fis+24] Raphael Fischer et al. “MetaQuRe: Meta-learning from Model Quality and Resource Consumption”. In: *Machine Learning and Knowledge Discovery in Databases*. 2024, pp. 209–226. ISBN: 978-3-031-70368-3. DOI: 10.1007/978-3-031-70368-3_13.

[Fis+25] Raphael Fischer et al. “Bridging the Communication Gap: Evaluating AI Labeling Practices for Trustworthy AI Development”. In: *Proceedings of the 2025 AAAI/ACM Conference on AI, Ethics, and Society*. (forthcoming). 2025. URL: https://arxiv.org/abs/2501.11909.

[FJM23] Raphael Fischer, Matthias Jakobs, and Katharina Morik. “Energy Efficiency Considerations for Popular AI Benchmarks”. In: *AI for Energy Innovation Workshop at the 37th AAAI Conference on Artificial Intelligence*. (forthcoming). 2023. URL: https://arxiv.org/abs/2304.08359.

[FLM24] Raphael Fischer, Thomas Liebig, and Katharina Morik. “Towards More Sustainable and Trustworthy Reporting in Machine Learning”. In: *Data Mining and Knowledge Discovery* (2024). ISSN: 1573-756X. DOI: 10.1007/s10618-024-01020-3.

[Flo+18] Luciano Floridi et al. “AI4People—An Ethical Framework for a Good AI Society: Opportunities, Risks, Principles, and Recommendations”. In: *Minds and Machines* (Dec. 2018), pp. 689–707. ISSN: 1572-8641. DOI: 10.1007/s11023-018-9482-5.

[FPM19] Raphael Fischer, Nico Piatkowski, and Katharina Morik. “Parameter Sharing for Spatio-Temporal Process Models”. In: *LWDA Workshops: FGWM, KDML, FGWI-BIA, and FGIR*. 2019, pp. 89–93. URL: https://ceur-ws.org/Vol-2454/paper_61.pdf.

[FS24] Raphael Fischer and Amal Saadallah. "AutoXPCR: Automated Multi-Objective Model Selection for Time Series Forecasting". In: *Proceedings of the 30th International Conference on Knowledge Discovery and Data Mining (KDD)*. 2024, pp. 806–815. ISBN: 979-8-4007-0490-1. DOI: 10.1145/3637528.3672057.

[FSB24] Raphael Fischer, Alexander van der Staay, and Sebastian Buschjäger. *Stress-Testing USB Accelerators for Efficient Edge Inference*. (preprint). 2024. DOI: 10.21203/rs.3.rs-3793927/v1.

[Gar+19] Eva García-Martín et al. "Estimation of Energy Consumption in Machine Learning". In: *Journal of Parallel and Distributed Computing* (Dec. 2019), pp. 75–88. ISSN: 0743-7315. DOI: 10.1016/j.jpdc.2019.07.007.

[Geb+21] Timnit Gebru et al. "Datasheets for Datasets". In: *Communications of the ACM* (Nov. 2021), pp. 86–92. ISSN: 0001-0782. DOI: 10.1145/3458723.

[Gho17] Tapabrata Ghosh. *QuickNet: Maximizing Efficiency and Efficacy in Deep Architectures*. 2017. URL: https://arxiv.org/abs/1701.02291.

[Gij+24] Pieter Gijsbers et al. "AMLB: An AutoML Benchmark". In: *Journal of Machine Learning Research* (2024), pp. 1–65. URL: https://jmlr.org/papers/v25/22-0493.html.

[Gio+24] Joseph Giovanelli et al. "Interactive Hyperparameter Optimization in Multi-Objective Problems via Preference Learning". In: *Proceedings of the AAAI Conference on Artificial Intelligence* (2024), pp. 12172–12180. URL: https://ojs.aaai.org/index.php/AAAI/article/view/29106.

[GLM+24] Team GLM et al. *ChatGLM: A Family of Large Language Models from GLM-130B to GLM-4 All Tools*. 2024. URL: https://arxiv.org/abs/2406.12793.

[GM22] Sergio Genovesi and Julia Maria Mönig. "Acknowledging Sustainability in the Framework of Ethical Certification for AI". In: *Sustainability* (2022). ISSN: 2071-1050. DOI: 10.3390/su14074157.

[GMP18] Ian Goodfellow, Patrick McDaniel, and Nicolas Papernot. "Making Machine Learning Robust Against Adversarial Inputs". In: *Communications of the ACM* (June 2018), pp. 56–66. ISSN: 0001-0782. DOI: 10.1145/3134599.

[God+21] Rakshitha Wathsadini Godahewa et al. "Monash Time Series Forecasting Archive". In: *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*. 2021. URL: https://openreview.net/forum?id=wEc1mgAjU-.

[Hac24] Philipp Hacker. "Sustainable AI Regulation". In: *Common Market Law Review* (2024), pp. 345–386. ISSN: 0165-0750. DOI: 10.54648/cola2024025.

[Hal+09] Mark Hall et al. "The WEKA Data Mining Software: An Update". In: *SIGKDD Explorations Newsletter* (Nov. 2009), pp. 10–18. ISSN: 1931-0145. DOI: 10.1145/1656274.1656278.

[Hal22] Aurélie Halsband. "Sustainable AI and Intergenerational Justice". In: *Sustainability* (2022). ISSN: 2071-1050. DOI: 10.3390/su14073922.

[Han+25] Matthew G. Hanna et al. “Future of Artificial Intelligence—Machine Learning Trends in Pathology and Medicine”. In: *Modern Pathology* (Apr. 2025), p. 100705. ISSN: 0893-3952. DOI: 10.1016/j.modpat.2025.100705.

[Har+20] Charles R. Harris et al. “Array Programming With NumPy”. In: *Nature* (Sept. 2020), pp. 357–362. DOI: 10.1038/s41586-020-2649-2.

[HD19] Dan Hendrycks and Thomas G. Dietterich. “Benchmarking Neural Network Robustness to Common Corruptions and Perturbations”. In: *7th International Conference on Learning Representations (ICLR)*. 2019. URL: https://openreview.net/forum?id=HJz6tiCqYm.

[He+16a] Kaiming He et al. “Deep Residual Learning for Image Recognition”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[He+16b] Kaiming He et al. “Identity Mappings in Deep Residual Networks”. In: *Proceedings of the 14th European Conference on Computer Vision (ECCV)*. 2016, pp. 630–645. DOI: 10.1007/978-3-319-46493-0_38.

[Hen+20] Peter Henderson et al. “Towards the Systematic Reporting of the Energy and Carbon Footprints of Machine Learning”. In: *Journal of Machine Learning Research* (2020), pp. 1–43. URL: https://jmlr.org/papers/v21/20-312.html.

[Hin+20] Michael Hind et al. “Experiences with Improving the Transparency of AI Models and Services”. In: *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*. 2020, pp. 1–8. ISBN: 978-1-4503-6819-3. DOI: 10.1145/3334480.3383051.

[HK06] Rob J. Hyndman and Anne B. Koehler. “Another Look at Measures of Forecast Accuracy”. In: *International Journal of Forecasting* (2006), pp. 679–688. ISSN: 0169-2070. DOI: 10.1016/j.ijforecast.2006.03.001.

[HKV19] Frank Hutter, Lars Kotthoff, and Joaquin Vanschoren. *Automated Machine Learning - Methods, Systems, Challenges*. Springer Nature, 2019. DOI: 10.1007/978-3-030-05318-5.

[HKZ23] Marc P. Hauer, Tobias D. Krafft, and Katharina Zweig. “Overview of Transparency and Inspectability Mechanisms to Achieve Accountability of Artificial Intelligence Systems”. In: *Data & Policy* (2023), e36. DOI: 10.1017/dap.2023.30.

[HM24] John Healy and Leland McInnes. “Uniform Manifold Approximation and Projection”. In: *Nature Reviews Methods Primers* (Nov. 2024), p. 82. ISSN: 2662-8449. DOI: 10.1038/s43586-024-00363-x.

[Hol+22] Andreas Holzinger et al. “Explainable AI Methods - A Brief Overview”. In: *xxAI - Beyond Explainable AI: International (ICML Workshop)*. 2022, pp. 13–38. ISBN: 978-3-031-04083-2. DOI: 10.1007/978-3-031-04083-2_2.

[Hol+23] Noah Hollmann et al. “TabPFN: A Transformer That Solves Small Tabular Classification Problems in a Second”. In: *11th International Conference on Learning Representations (ICLR)*. 2023. URL: https://openreview.net/forum?id=cp5PvcI6w8_.

[How+17] Andrew Howard et al. *Mobilenets: Efficient Convolutional Neural Networks for Mobile Vision Applications*. 2017. URL: https://arxiv.org/abs/1704.04861.

[How+19] Andrew Howard et al. “Searching for MobileNetV3”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Oct. 2019. DOI: 10.1109/ICCV.2019.00140.

[HTF09] Trevor Hastie, Robert Tibshirani, and Jerome H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd Edition*. Springer, 2009. ISBN: 9780387848570. DOI: 10.1007/978-0-387-84858-7.

[Hua+17] Gao Huang et al. “Densely Connected Convolutional Networks”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 2261–2269. DOI: 10.1109/CVPR.2017.243.

[Hua24] Jensen Huang. *Interview on the ”No Priors” Podcast*. Accessed: 2025-08-13. Nov. 2024. URL: https://youtu.be/hw7EnjC68Fw.

[Hug24] Hugging Face, Inc. *Hugging Face: AI Community and Tools*. Accessed: 2025-08-13. 2024. URL: https://huggingface.co.

[Hut18] Matthew Hutson. “Artificial Intelligence Faces Reproducibility Crisis”. In: *Science* (Feb. 2018), pp. 725–726. DOI: 10.1126/science.359.6377.725.

[Int22] Intel. *Intel Neural Compute Stick 2 Discontinuation Notice*. Accessed: 2025-08-13. 2022. URL: https://www.intel.com/content/www/us/en/support/articles/000093181/boards-and-kits.html.

[Int24a] Intel Corporation. *Intel® 64 and IA-32 Architectures Software Developer’s Manual, Volume 3: System Programming Guide*. Accessed: 2025-08-13. 2024. URL: https://intel.com/content/www/us/en/developer/articles/technical/intel-sdm.html.

[Int24b] International Telecommunication Union. *Disaster Management: The Standards Perspective*. Accessed: 2025-08-13. 2024. URL: https://aiforgood.itu.int/newsroom/publications-and-reports/.

[Isl+19] Md Johirul Islam et al. “A Comprehensive Study on Deep Learning Bug Characteristics”. In: *Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 2019, pp. 510–520. ISBN: 978-1-4503-5572-8. DOI: 10.1145/3338906.3338955.

[JCQ23] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. *Ultralytics YOLOv8*. Version 8.0.0. 2023. URL: https://github.com/ultralytics/ultralytics.

[Jev65] William Stanley Jevons. *The Coal Question: An Enquiry Concerning the Progress of the Nation, and the Probable Exhaustion of Our Coal-mines*. Macmillan, 1865. ISBN: 978-1-78987-646-8. URL: https://books.google.de/books?id=gAAKAAAAIAAJ.

[JIV19] Anna Jobin, Marcello Ienca, and Effy Vayena. "The Global Landscape of AI Ethics Guidelines". In: *Nature Machine Intelligence* (Sept. 2019), pp. 389–399. ISSN: 2522-5839. DOI: 10.1038/s42256-019-0088-2.

[JM25] Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models*. 3rd. Online manuscript. 2025. URL: https://web.stanford.edu/~jurafsky/slp3/.

[JSG21] Hadi S. Jomaa, Lars Schmidt-Thieme, and Josif Grabocka. "Dataset2Vec: Learning Dataset Meta-Features". In: *Data Mining and Knowledge Discovery* (May 2021), pp. 964–985. ISSN: 1573-756X. DOI: 10.1007/s10618-021-00737-9.

[JSH19] Haifeng Jin, Qingquan Song, and Xia Hu. "Auto-Keras: An Efficient Neural Architecture Search System". In: *Proceedings of the 25th International Conference on Knowledge Discovery and Data Mining (KDD)*. 2019, pp. 1946–1956. ISBN: 9781450362016. DOI: 10.1145/3292500.3330648.

[JT22] David T. Jones and Janet M. Thornton. "The Impact of AlphaFold2 One Year On". In: *Nature Methods* (Jan. 2022), pp. 15–20. ISSN: 1548-7105. DOI: 10.1038/s41592-021-01365-3.

[Kag24] Kaggle, Inc. *Kaggle: Your Machine Learning and Data Science Community*. Accessed: 2025-08-13. 2024. URL: https://kaggle.com.

[KCS22] Arpan Kumar Kar, Shweta Kumari Choudhary, and Vinay Kumar Singh. "How Can Artificial Intelligence Impact Sustainability: A Systematic Literature Review". In: *Journal of Cleaner Production* (2022), p. 134120. DOI: 10.1016/j.jclepro.2022.134120.

[KFR08] Dan J. Kim, Donald L. Ferrin, and H. Raghav Rao. "A Trust-Based Consumer Decision-Making Model in Electronic Commerce: The Role of Trust, Perceived Risk, and Their Antecedents". In: *Decision support systems* (2008), pp. 544–564. DOI: 10.1016/j.dss.2007.07.001.

[Kha+18] Jayden Khakurel et al. "The Rise of Artificial Intelligence under the Lens of Sustainability". In: *Technologies* (2018). ISSN: 2227-7080. DOI: 10.3390/technologies6040100.

[Kim+16] Dan J. Kim et al. "Web Assurance Seal Services, Trust and Consumers' Concerns: An Investigation of E-commerce Transaction Intentions Across Two Nations". In: *European Journal of Information Systems* (2016), pp. 252–273. DOI: 10.1057/ejis.2015.16.

[KJ95] Ron Kohavi and George H. John. "Autmatic Parameter Selection by Minimizing Estimated Error". In: *Proceedings of the 12th International Conference on Machine Learning (ICML)*. 1995, pp. 304–312. ISBN: 1558603778. DOI: 10.1016/B978-1-55860-377-6.50045-1.

[KSH12] Iacovos Kirlappos, M. Angela Sasse, and Nigel Harvey. "Why Trust Seals Don't Work: A Study of User Perceptions and Behavior". In: *Trust and Trustworthy Computing*. 2012, pp. 308–324. ISBN: 978-3-642-30921-2. DOI: 10.1007/978-3-642-30921-2_18.

[KSH17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. "ImageNet Classification with Deep Convolutional Neural Networks". In: *Communications of the ACM* (May 2017), pp. 84–90. ISSN: 0001-0782. DOI: 10.1145/3065386.

[Küh+20] Niklas Kühl et al. "How to Conduct Rigorous Supervised Machine Learning in Information Systems Research: The Supervised Machine Learning Report Card". In: *Communications of the Association for Information Systems* (Dec. 2020). DOI: 10.17705/1CAIS.04845.

[Kus+17] Matt J. Kusner et al. "Counterfactual Fairness". In: *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*. 2017. URL: https://dl.acm.org/doi/10.5555/3294996.3295162.

[KWI24] Nils Körber, Silvan Wehrli, and Christopher Irrgang. *How to Measure the Intelligence of Large Language Models?* 2024. URL: https://arxiv.org/abs/2407.20828.

[Lak+17] Brenden M. Lake et al. "Building Machines That Learn and Think Like People". In: *Behavioral and Brain Sciences* (2017). DOI: 10.1017/S0140525X16001837.

[Lam25] Lamarr Institute for Machine Learning and Artificial Intelligence. *Research on ML and AI*. Accessed: 2025-08-13. 2025. URL: https://lamarr-institute.org/research/.

[Lan+21] Markus Langer et al. "What Do We Want From Explainable Artificial Intelligence (Xai)? – A Stakeholder Perspective on Xai and a Conceptual Model Guiding Interdisciplinary Xai Research". In: *Artificial Intelligence* (2021), p. 103473. ISSN: 0004-3702. DOI: 10.1016/j.artint.2021.103473.

[Lar+96] Jan Larsen et al. "Design and Regularization of Neural Networks: The Optimal Use of a Validation Set". In: *Neural Networks for Signal Processing VI. Proceedings of the 1996 IEEE Signal Processing Society Workshop*. 1996, pp. 62–71. DOI: 10.1109/NNSP.1996.548336.

[LBH15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. "Deep Learning". In: *Nature* (May 2015), pp. 436–444. ISSN: 1476-4687. DOI: 10.1038/nature14539.

[LD22] Zimeng Lyu and Travis Desell. "ONE-NAS: An Online Neuroevolution Based Neural Architecture Search for Time Series Forecasting". In: *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. 2022, pp. 659–662. ISBN: 9781450392686. DOI: 10.1145/3520304.3528962.

[Ley02] Michael Ley. "The DBLP Computer Science Bibliography: Evolution, Research Issues, Perspectives". In: *Proceedings of the 9th International Symposium on String Processing and Information Retrieval (SPIRE)*. 2002, pp. 1–10. DOI: 10.1007/3-540-45735-6_1.

[Lia+24] Weixin Liang et al. "Systematic Analysis of 32,111 AI Model Cards Characterizes Documentation Practice in AI". In: *Nature Machine Intelligence* (July 2024), pp. 744–753. ISSN: 2522-5839. DOI: 10.1038/s42256-024-00857-z.

[Lim+21] Bryan Lim et al. "Temporal Fusion Transformers for Interpretable Multi-Horizon Time Series Forecasting". In: *International Journal of Forecasting* (2021), pp. 1748–1764. DOI: 10.1016/j.ijforecast.2021.03.012.

[Lin+21] Sebastian Lins et al. "Artificial Intelligence as a Service". In: *Business & Information Systems Engineering* (Aug. 2021), pp. 441–456. ISSN: 1867-0202. DOI: 10.1007/s12599-021-00708-w.

[LJS24] Alexandra Sasha Luccioni, Yacine Jernite, and Emma Strubell. "Power Hungry Processing: Watts Driving the Cost of AI Deployment?" In: *Proceedings of the 7th Conference on Fairness, Accountability and Transparency (FAccT)*. 2024, pp. 85–99. ISBN: 9798400704505. DOI: 10.1145/3630106.3658542.

[Lon+24] Luca Longo et al. "Explainable Artificial Intelligence (XAI) 2.0: A Manifesto of Open Challenges and Interdisciplinary Research Directions". In: *Information Fusion* (June 2024), p. 102301. ISSN: 1566-2535. DOI: 10.1016/j.inffus.2024.102301.

[LS04] John D. Lee and Katrina A. See. "Trust in Automation: Designing for Appropriate Reliance". In: *Human Factors* (Mar. 2004), pp. 50–80. ISSN: 0018-7208. DOI: 10.1518/hfes.46.1.50_30392.

[LS16] Jens Lansing and Ali Sunyaev. "Trust in Cloud Computing: Conceptual Typology and Trust-Building Antecedents". In: *SIGMIS Database* (June 2016), pp. 58–96. ISSN: 0095-0033. DOI: 10.1145/2963175.2963179.

[LS19] Jobst Landgrebe and Barry Smith. *There is no Artificial General Intelligence*. 2019. URL: https://arxiv.org/abs/1906.05833.

[LSC25] Alexandra Sasha Luccioni, Emma Strubell, and Kate Crawford. *From Efficiency Gains to Rebound Effects: The Problem of Jevons' Paradox in AI's Polarized Environmental Debate*. 2025. URL: https://arxiv.org/abs/2501.16548.

[LTM24] Alexandra Sasha Luccioni, Bruna Trevelin, and Margaret Mitchell. *The Environmental Impacts of AI – Policy Primer*. Hugging Face Blog. 2024. DOI: 10.57967/hf/3004.

[Luc+19] Alexandra Sasha Luccioni et al. "Quantifying the Carbon Emissions of Machine Learning". In: *NeurIPS Workshop on Tackling Climate Change with Machine Learning*. 2019. URL: https://climatechange.ai/papers/neurips2019/22.

[Luc+25] Alexandra Sasha Luccioni et al. *AI Energy Score Documentation*. Accessed: 2025-08-13. 2025. URL: https://huggingface.github.io/AIEnergyScore.

[LVL23] Alexandra Sasha Luccioni, Sylvain Viguier, and Anne-Laure Ligozat. "Estimating the Carbon Footprint of BLOOM, a 176B Parameter Language Model". In: *Journal of Machine Learning Research* (2023), pp. 1–15. URL: https://jmlr.org/papers/v24/23-0069.html.

[LZJ22] Shiqing Liu, Haoyu Zhang, and Yaochu Jin. "A Survey on Computationally Efficient Neural Architecture Search". In: *Journal of Automation and Intelligence* (2022), p. 100002. ISSN: 2949-8554. DOI: 10.1016/j.jai.2022.100002.

[MBC24] Barbara Martini, Denise Bellisario, and Paola Coletti. "Human-Centered and Sustainable Artificial Intelligence in Industry 5.0: Challenges and Perspectives". In: *Sustainability* (2024). ISSN: 2071-1050. DOI: 10.3390/su16135448.

[MC04] Pamela McCorduck and Cli Cfe. *Machines Who Think: A Personal Inquiry into the History and Prospects of Artificial Intelligence*. AK Peters/CRC Press, 2004. ISBN: 978-0-429-25898-5. DOI: 10.1201/9780429258985.

[McC+55] John McCarthy et al. *A Proposal for The Dartmouth Summer Research Project on Artificial Intelligence*. 1955. DOI: 10.1609/aimag.v27i4.1904.

[Mck+11] D Harrison Mcknight et al. "Trust in a Specific Technology: An Investigation of its Components and Measures". In: *Transactions on Management Information Systems* (2011), pp. 1–25. DOI: 10.1145/1985347.1985353.

[McK10] Wes McKinney. "Data Structures for Statistical Computing in Python". In: *Proceedings of the 9th Python in Science Conference*. 2010, pp. 56–61. DOI: 10.25080/Majora-92bf1922-00a.

[Meh+21] Ninareh Mehrabi et al. "A Survey on Bias and Fairness in Machine Learning". In: *ACM Computing Surveys* (July 2021). ISSN: 0360-0300. DOI: 10.1145/3457607.

[Mit+19] Margaret Mitchell et al. "Model Cards for Model Reporting". In: *Proceedings of the 2nd Conference on Fairness, Accountability and Transparency (FAccT)*. 2019, pp. 220–229. ISBN: 978-1-4503-6125-5. DOI: 10.1145/3287560.3287596.

[Mök+22] Jakob Mökander et al. "The US Algorithmic Accountability Act of 2022 vs. The EU Artificial Intelligence Act: what can they learn from each other?" In: *Minds and Machines* (Dec. 2022), pp. 751–758. ISSN: 1572-8641. DOI: 10.1007/s11023-022-09612-y.

[Moo98] Gordon E. Moore. "Cramming More Components Onto Integrated Circuits". In: *Proceedings of the IEEE* (Jan. 1998), pp. 82–85. ISSN: 1558-2256. DOI: 10.1109/JPROC.1998.658762.

[Mor+21] Katharina Morik et al. *The Care Label Concept: A Certification Suite for Trustworthy and Resource-Aware Machine Learning*. 2021. URL: https://arxiv.org/abs/2106.00512.

[Mor+22] Katharina Morik et al. “Yes We Care! - Certification for Machine Learning Methods Through the Care Label Framework”. In: *Frontiers in Artificial Intelligence* (2022). DOI: 10.3389/frai.2022.975029.

[MP23] Ilyas Moutawwakil and Régis Pierrard. *LLM-Perf Leaderboard*. Accessed: 2025-08-13. 2023. URL: https://huggingface.co/spaces/optimum/llm-perf-leaderboard.

[MP43] Warren S. McCulloch and Walter Pitts. “A Logical Calculus of the Ideas Immanent in Nervous Activity”. In: *The bulletin of mathematical biophysics* (Dec. 1943), pp. 115–133. ISSN: 1522-9602. DOI: 10.1007/BF02478259.

[MT23] Wayne Moodaley and Arnesh Telukdarie. “Greenwashing, Sustainability Reporting, and Artificial Intelligence: A Systematic Literature Review”. In: *Sustainability* (2023). ISSN: 2071-1050. DOI: 10.3390/su15021481.

[Mul+18] Jean-Michel Muller et al. *Handbook of Floating-Point Arithmetic*. Springer, 2018. ISBN: 978-3-319-76525-9. DOI: 10.1007/978-3-319-76526-6.

[MW07] Poornima Madhavan and Douglas Wiegmann. “Similarities and Differences Between Human–Human and Human–Automation Trust: An Integrative Review”. In: *Theoretical Issues in Ergonomics Science* (2007), pp. 277–301. ISSN: 1463-922X. DOI: 10.1080/14639220500337708.

[MW23] Felix Mohr and Marcel Wever. “Naive Automated Machine Learning”. In: *Machine Learning* (Apr. 2023), pp. 1131–1170. ISSN: 1573-0565. DOI: 10.1007/s10994-022-06200-0.

[Nah+23] Nadia Nahar et al. “A Meta-Summary of Challenges in Building Products with ML Components – Collecting Experiences from 4758+ Practitioners”. In: *Proceedings of the 2nd International Conference on AI Engineering - Software Engineering for AI (CAIN)*. May 2023, pp. 171–183. DOI: 10.1109/CAIN58948.2023.00034.

[Nau+23] Meike Nauta et al. “From Anecdotal Evidence to Quantitative Evaluation Methods: A Systematic Review on Evaluating Explainable AI”. In: *ACM Computing Surveys* (July 2023). ISSN: 0360-0300. DOI: 10.1145/3583558.

[New23] Gavin Newsom. *Generative AI (GenAI) Executive Order*. Accessed: 2025-08-13. 2023. URL: https://www.govops.ca.gov/generative-ai-genai-executive-order/.

[Ng+21] Davy Tsz Kit Ng et al. “Conceptualizing AI Literacy: An Exploratory Review”. In: *Computers and Education: Artificial Intelligence* (2021), p. 100041. ISSN: 2666-920X. DOI: 10.1016/j.caeai.2021.100041.

[NLA25] Felix Neutatz, Marius Lindauer, and Ziawasch Abedjan. “How Green is AutoML for Tabular Data?” In: *Proceedings of the 28th International Conference on Extending Database Technology (EDBT)*. 2025, pp. 350–363. DOI: 10.48786/EDBT.2025.28.

[NTF24] Claudio Novelli, Mariarosaria Taddeo, and Luciano Floridi. "Accountability in Artificial Intelligence: What It Is and How It Works". In: *AI & SOCIETY* (Aug. 2024), pp. 1871–1882. ISSN: 1435-5655. DOI: 10.1007/s00146-023-01635-y.

[NVI12] NVIDIA Corporation. *NVIDIA Management Library (NVML)*. Accessed: 2025-08-13. 2012. URL: https://developer.nvidia.com/management-library-nvml.

[NZS23] Gireen Naidu, Tranos Zuva, and Elias Mmbongeni Sibanda. "A Review of Evaluation Metrics in Machine Learning Algorithms". In: *Artificial Intelligence Application in Networks and Systems*. 2023, pp. 15–25. ISBN: 978-3-031-35314-7. DOI: 10.1007/978-3-031-35314-7_2.

[Oor+16] Aäron van den Oord et al. "WaveNet: A Generative Model for Raw Audio". In: *International Symposium on Computer Architecture (ISCA) Speech Synthesis Workshop*. 2016, p. 125. DOI: 10.48550/arXiv.1609.03499.

[Ope+24] OpenAI et al. *GPT-4 Technical Report*. 2024. URL: https://arxiv.org/abs/2303.08774.

[Ore+20] Boris N. Oreshkin et al. "N-BEATS: Neural Basis Expansion Analysis for Interpretable Time Series Forecasting". In: *8th International Conference on Learning Representations (ICLR)*. 2020. URL: https://openreview.net/forum?id=r1ecqn4YwB.

[Org22] BEUC – The European Consumer Organisation. *Revision of EU Legislation on Food Information To Consumers*. Accessed: 2025-08-13. 2022. URL: https://www.beuc.eu/food-labelling-healthier-choices-towards-eu-wide-nutri-score.

[Pap24] Papers with Code. *Papers With Code: State-of-the-Art Machine Learning Papers, Code, and Evaluation Tables*. Accessed: 2025-08-13. 2024. URL: https://paperswithcode.com.

[Pat+22] David Patterson et al. "The Carbon Footprint of Machine Learning Training Will Plateau, Then Shrink". In: *Computer* (2022), pp. 18–28. ISSN: 1558-0814. DOI: 10.1109/MC.2022.3148714.

[PE24] European Parliament and Council of the European Union. *Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence*. Official Journal of the European Union, accessed: 2025-08-13. 2024. URL: https://data.europa.eu/eli/reg/2024/1689/oj.

[Ped+11] Fabian Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research (JMLR)* (2011), pp. 2825–2830. DOI: 10.5555/1953048.2078195.

[Pei+19] Kexin Pei et al. "DeepXplore: Automated Whitebox Testing of Deep Learning Systems". In: *Communications of the ACM* (Oct. 2019), pp. 137–145. ISSN: 0001-0782. DOI: 10.1145/3361566.

[Pia19] Nico Philipp Piatkowski. "Exponential Families on Resource-Constrained Systems". PhD thesis. TU Dortmund University, 2019. DOI: 10.17877/DE290R-18876.

[Pio+21] David Piorkowski et al. "How AI Developers Overcome Communication Challenges in a Multidisciplinary Team: A Case Study". In: *Proceedings of the ACM on Human-Computer Interaction* (Apr. 2021). DOI: 10.1145/3449205.

[Plo24] Plotly Technologies Inc. *Plotly: Collaborative Data Science and Visualization*. Accessed: 2025-08-13. 2024. URL: https://plotly.com.

[PPP24] Sergiusz Pimenow, Olena Pimenowa, and Piotr Prus. "Challenges of Artificial Intelligence Development in the Context of Energy Consumption and Impact on Climate Change". In: *Energies* (2024). ISSN: 1996-1073. DOI: 10.3390/en17235965.

[Rad+23] Alec Radford et al. "Robust Speech Recognition via Large-Scale Weak Supervision". In: *Proceedings of the 40th International Conference on Machine Learning (ICML)*. July 2023, pp. 28492–28518. URL: https://proceedings.mlr.press/v202/radford23a.html.

[Ran+18] Syama Sundar Rangapuram et al. "Deep State Space Models for Time Series Forecasting". In: *Proceedings of the 32nd International Conference on Neural Information Processing Systems (NeurIPS)*. 2018, pp. 7796–7805. URL: https://dl.acm.org/doi/10.5555/3327757.3327876.

[RDI23] Oana-Marina Radu, Voicu D. Dragomir, and Liliana Ionescu-Feleagă. "The Link between Corporate ESG Performance and the UN Sustainable Development Goals". In: *Proceedings of the International Conference on Business Excellence* (2023), pp. 776–790. DOI: 10.2478/picbe-2023-0072.

[Ren25] Render. *Render: Cloud Application Platform Documentation*. Accessed: 2025-08-13. 2025. URL: https://render.com/docs.

[Rep24] ReportLab, Inc. *ReportLab: Open-Source Python PDF Library*. Accessed: 2025-08-13. 2024. URL: https://reportlab.com.

[Reu+19] Albert Reuther et al. "Survey and Benchmarking of Machine Learning Accelerators". In: *Proceedings of the 23rd High Performance Extreme Computing Conference (HPEC)*. Sept. 2019, pp. 1–9. DOI: 10.1109/HPEC.2019.8916327.

[RN21] Stuart Russell and Peter Norvig. *Artificial Intelligence, Global Edition : A Modern Approach*. Pearson Deutschland, May 2021. ISBN: 978-1292401133. URL: https://elibrary.pearson.de/book/99.150005/9781292401171.

[Roh+24] Friederike Rohde et al. "Broadening the Perspective for Sustainable Artificial Intelligence: Sustainability Criteria and Indicators for Artificial Intelligence Systems". In: *Current Opinion in Environmental Sustainability* (Feb. 2024), p. 101411. ISSN: 1877-3435. DOI: 10.1016/j.cosust.2023.101411.

[Ros58] Frank Rosenblatt. “The Perceptron: A Probabilistic Model For Information Storage And Organization in the Brain”. In: *Psychological Review* (1958), pp. 386–408. ISSN: 1939-1471. DOI: 10.1037/h0042519.

[Rot24] Emma Roth. *ChatGPT Now Has Over 300 Million Weekly Users*. Accessed: 2025-08-13. 2024. URL: https://theverge.com/2024/12/4/24313097/chatgpt-300-million-weekly-users.

[Rud19] Cynthia Rudin. “Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead”. In: *Nature Machine Intelligence* (May 2019), pp. 206–215. ISSN: 2522-5839. DOI: 10.1038/s42256-019-0048-x.

[Rue+23] Laura von Rueden et al. “Informed Machine Learning – A Taxonomy and Survey of Integrating Prior Knowledge into Learning Systems”. In: *Transactions on Knowledge and Data Engineering* (2023), pp. 614–633. DOI: 10.1109/TKDE.2021.3079836.

[Rut+24] Jérôme Rutinowski et al. “Benchmarking Trust: A Metric for Trustworthy Machine Learning”. In: *Explainable Artificial Intelligence*. 2024, pp. 287–307. ISBN: 978-3-031-63787-2. DOI: 10.1007/978-3-031-63787-2_15.

[Sæt21] Henrik S. Sætra. “AI in Context and the Sustainable Development Goals: Factoring in the Unsustainability of the Sociotechnical System”. In: *Sustainability* (2021). ISSN: 2071-1050. DOI: 10.3390/su13041738.

[Sak+21] Keisuke Sakaguchi et al. “WinoGrande: an adversarial winograd schema challenge at scale”. In: *Communications of the ACM* (Aug. 2021), pp. 99–106. ISSN: 0001-0782. DOI: 10.1145/3474381.

[Sal+20] David Salinas et al. “DeepAR: Probabilistic forecasting with autoregressive recurrent networks”. In: *International Journal of Forecasting* (2020), pp. 1181–1191. DOI: 10.1016/j.ijforecast.2019.07.001.

[Sal+25] Malik Sallam et al. “DeepSeek: Is it the End of Generative AI Monopoly or the Mark of the Impending Doomsday?” In: *Mesopotamian Journal of Big Data* (2025). Section: Articles, pp. 26–34. DOI: 10.58496/MJBD/2025/002.

[Sam+19] Wojciech Samek et al. *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Springer Nature, 2019. DOI: 10.1007/978-3-030-28954-6.

[Sam59] Arthur L. Samuel. “Some Studies in Machine Learning Using the Game of Checkers”. In: *IBM Journal of Research and Development* (July 1959), pp. 210–229. ISSN: 0018-8646. DOI: 10.1147/rd.33.0210.

[San+18] Mark Sandler et al. “MobileNetV2: Inverted Residuals and Linear Bottlenecks”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018, pp. 4510–4520. DOI: 10.1109/CVPR.2018.00474.

[Sch+20] Roy Schwartz et al. “Green AI”. In: *Communications of the ACM* (Nov. 2020), pp. 54–63. ISSN: 0001-0782. DOI: 10.1145/3381831.

[Sch+22] Anna Schmitz et al. “The Why and How of Trustworthy AI: An Approach for Systematic Quality Assurance When Working With ML Components”. In: *at - Automatisierungstechnik* (2022), pp. 793–804. DOI: 10.1515/auto-2022-0012.

[Sch+23] Nicolas Scharowski et al. “Certification Labels for Trustworthy AI: Insights From an Empirical Mixed-Method Study”. In: *Proceedings of the 6th Conference on Fairness, Accountability and Transparency (FAccT)*. 2023, pp. 248–260. ISBN: 979-8-4007-0192-4. DOI: 10.1145/3593013.3593994.

[Sch87] Jürgen Schmidhuber. “Evolutionary Principles in Self-Referential Learning, or on Learning How to Learn: The Meta-Meta-... Hook”. en. PhD thesis. Technische Universität München, 1987. URL: https://mediatum.ub.tum.de/?id=813180.

[Sev+22] Jaime Sevilla et al. “Compute Trends Across Three Eras of Machine Learning”. In: *2022 International Joint Conference on Neural Networks (IJCNN)*. 2022, pp. 1–8. DOI: 10.1109/IJCNN55064.2022.9891914.

[SF20] Kacper Sokol and Peter Flach. “One Explanation Does Not Fit All”. In: *KI - Künstliche Intelligenz* (June 2020), pp. 235–250. ISSN: 1610-1987. DOI: 10.1007/s13218-020-00637-y.

[SFB24] Alexander van der Staay, Raphael Fischer, and Sebastian Buschjäger. “Stress-Testing USB Accelerators for Efficient Edge Inference”. In: *Proceedings of the 9th Symposium on Edge Computing (SEC)*. 2024, pp. 1–14. DOI: 10.1109/SEC62691.2024.00015.

[SGM20] Emma Strubell, Ananya Ganesh, and Andrew McCallum. “Energy and Policy Considerations for Modern Deep Learning Research”. In: *Proceedings of the AAAI Conference on Artificial Intelligence* (Apr. 2020), pp. 13693–13696. DOI: 10.1609/aaai.v34i09.7123.

[Sha+16] Bobak Shahriari et al. “Taking the Human Out of the Loop: A Review of Bayesian Optimization”. In: *Proceedings of the IEEE* (2016), pp. 148–175. DOI: 10.1109/JPROC.2015.2494218.

[Sha+24] Ayyoob Sharifi et al. “Smart Cities and Sustainable Development Goals (SDGs): A Systematic Literature Review of Co-benefits and Trade-Offs”. In: *Cities* (Mar. 2024), p. 104659. ISSN: 0264-2751. DOI: 10.1016/j.cities.2023.104659.

[Shc+23] Oleksandr Shchur et al. “AutoGluon–TimeSeries: AutoML for Probabilistic Time Series Forecasting”. In: *Proceedings of the 2nd International Conference on Automated Machine Learning*. Nov. 2023, pp. 9/1–21. URL: https://proceedings.mlr.press/v224/shchur23a.html.

[Shn20] Ben Shneiderman. “Human-Centered Artificial Intelligence: Reliable, Safe & Trustworthy”. In: *International Journal of Human–Computer Interaction* (Apr. 2020), pp. 495–504. ISSN: 1044-7318. DOI: 10.1080/10447318.2020.1741118.

[SJ21] Mirka Saarela and Susanne Jauhiainen. "Comparison of Feature Importance Measures as Explanations for Classification Models". In: *Discover Applied Sciences* (Feb. 2021), p. 272. ISSN: 2523-3971. DOI: 10.1007/s42452-021-04148-9.

[SJH22] Q. Song, H. Jin, and X. Hu. *Automated Machine Learning in Action*. Manning, 2022. ISBN: 978-1-61729-805-9. URL: https://books.google.de/books?id=ZCZoEAAAQBAJ.

[SJM22] Amal Saadallah, Matthias Jakobs, and Katharina Morik. "Explainable Online Ensemble of Deep Neural Network Pruning for Time Series Forecasting". In: *Machine Learning* (Sept. 2022), pp. 3459–3487. ISSN: 1573-0565. DOI: 10.1007/s10994-022-06218-4.

[SLA12] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. "Practical Bayesian Optimization of Machine Learning Algorithms". In: *Proceedings of the 26th International Conference on Neural Information Processing Systems (NeurIPS)*. 2012. URL: https://dl.acm.org/doi/10.5555/2999325.2999464.

[Sol24] John Soldatos. *Artificial Intelligence in Manufacturing: Enabling Intelligent, Flexible and Cost-Effective Production Through AI*. Springer Nature, 2024. ISBN: 978-3-031-46451-5. URL: https://link.springer.com/book/10.1007/978-3-031-46452-2.

[SP18] Andrew Selbst and Julia Powles. ""Meaningful Information" and the Right to Explanation". In: *Proceedings of the 1st Conference on Fairness, Accountability and Transparency (FAccT)*. Feb. 2018, pp. 48–48. URL: https://proceedings.mlr.press/v81/selbst18a.html.

[Spe22] Timo Speith. "A Review of Taxonomies of Explainable Artificial Intelligence (XAI) Methods". In: *Proceedings of the 5th Conference on Fairness, Accountability and Transparency (FAccT)*. 2022, pp. 2239–2250. ISBN: 978-1-4503-9352-2. DOI: 10.1145/3531146.3534639.

[SRA24] Daniel Schönle, Christoph Reich, and Djaffar Ould Abdeslam. "Streamlining AI: Techniques for Efficient Machine Learning Model Selection". In: *The International Journal on Advances in Intelligent Systems* (June 2024), pp. 73–87. ISSN: 1942-2679. URL: https://personales.upv.es/thinkmind/IntSys/IntSys_v17_n12_2024/.

[Sri+23] Aarohi Srivastava et al. "Beyond the Imitation Game: Quantifying and extrapolating the capabilities of language models". In: *Transactions on Machine Learning Research* (2023). ISSN: 2835-8856. URL: https://openreview.net/forum?id=uyTL5Bvosj.

[SSW19] Christin Seifert, Stefanie Scherzinger, and Lena Wiese. "Towards Generating Consumer Labels for Machine Learning Models". In: *Proceedings of the 1st International Conference on Cognitive Machine Intelligence (CogMI)*. 2019, pp. 173–179. DOI: 10.1109/CogMI48466.2019.00033.

[Sta+25] Alexander van der Staay et al. “Reflective Design Theorizing with User Interviews: A Case Study for AI Energy Labels”. In: *Local Solutions for Global Challenges*. 2025, pp. 66–80. ISBN: 978-3-031-93979-2. DOI: 10.1007/978-3-031-93979-2_5.

[Sta24] State of Tennessee. *Ensuring Likeness, Voice, and Image Security (ELVIS) Act*. Accessed: 2025-04-11. 2024. URL: https://capitol.tn.gov/.

[Sut+13] Ilya Sutskever et al. “On the Importance of Initialization and Momentum in Deep Learning”. In: *Proceedings of the 30th International Conference on Machine Learning (ICML)*. 2013, pp. 1139–1147. URL: https://proceedings.mlr.press/v28/sutskever13.html.

[SZ15] Karen Simonyan and Andrew Zisserman. “Very Deep Convolutional Networks for Large-Scale Image Recognition”. In: *3rd International Conference on Learning Representations (ICLR)*. 2015. URL: https://arxiv.org/abs/1409.1556.

[Sze11] Richard Szeliski. *Computer Vision: Algorithms and Applications*. 2nd. Springer-Verlag, 2011. ISBN: 978-1-84882-935-0. DOI: 10.1007/978-3-030-34372-9.

[tea20] The pandas development team. *pandas-dev/pandas: Pandas*. Feb. 2020. DOI: 10.5281/zenodo.3509134.

[Tho+13] Chris Thornton et al. “Auto-WEKA: Combined Selection and Hyperparameter Optimization of Classification Algorithms”. In: *Proceedings of the 19th International Conference on Knowledge Discovery and Data Mining (KDD)*. 2013, pp. 847–855. ISBN: 9781450321747. DOI: 10.1145/2487575.2487629.

[TL19] Mingxing Tan and Quoc Le. “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks”. In: *Proceedings of the 36th International Conference on Machine Learning (ICML)*. June 2019, pp. 6105–6114. URL: https://proceedings.mlr.press/v97/tan19a.html.

[Tor+23] Tanja Tornede et al. “Towards Green Automated Machine Learning: Status Quo and Future Directions”. In: *Journal of Artificial Intelligence Research* (2023), pp. 427–457. DOI: 10.1613/JAIR.1.14340.

[TPV09] Grigorios Tsoumakas, Ioannis Partalas, and Ioannis Vlahavas. “An Ensemble Pruning Primer”. In: *Applications of Supervised and Unsupervised Ensemble Methods*. 2009, pp. 1–13. ISBN: 978-3-642-03999-7. DOI: 10.1007/978-3-642-03999-7_1.

[Tru20] Jon Truby. “Governing Artificial Intelligence to Benefit the UN Sustainable Development Goals”. In: *Sustainable Development* (July 2020), pp. 946–959. ISSN: 0968-0802. DOI: 10.1002/sd.2048.

[Tür+21] Ali Caner Türkmen et al. “Forecasting Intermittent and Sparse Time Series: A Unified Probabilistic Framework via Deep Renewal Processes”. In: *PLoS ONE* (2021), e0259764. DOI: 10.1371/journal.pone.0259764.

[Tur+24] Tommaso Turchi et al. “Pathways to Democratized Healthcare: Envisioning Human-Centered AI-As-A-Service for Customized Diagnosis and Rehabilitation”. In: *Artificial Intelligence in Medicine* (May 2024), p. 102850. ISSN: 0933-3657. DOI: 10.1016/j.artmed.2024.102850.

[Tur50] Alan M. Turing. “Computing Machinery and Intelligence”. In: *Mind* (Oct. 1950), pp. 433–460. ISSN: 0026-4423. DOI: 10.1093/mind/LIX.236.433.

[Uni22] United States Congress. *Algorithmic Accountability Act of 2022*. Accessed: 2025-08-13. 2022. URL: https://congress.gov/bill/117th-congress/senate-bill/3572.

[Uta24] Utah State Legislature. *Utah Artificial Intelligence Policy Act*. Accessed: 2025-08-13. 2024. URL: https://le.utah.gov/~2024/bills/static/SB0149.html.

[Vad15] Siddhartha Vadlamudi. “Enabling Trustworthiness in Artificial Intelligence - A Detailed Discussion”. In: *Engineering International* (Dec. 2015), pp. 105–114. DOI: 10.18034/ei.v3i2.519.

[Van+14] Joaquin Vanschoren et al. “OpenML: Networked Science in Machine Learning”. In: *SIGKDD Explorations Newsletter* (June 2014), pp. 49–60. ISSN: 1931-0145. DOI: 10.1145/2641190.2641198.

[Van19] Joaquin Vanschoren. “Meta-Learning”. In: *Automated Machine Learning: Methods, Systems, Challenges*. 2019, pp. 35–61. ISBN: 978-3-030-05318-5. DOI: 10.1007/978-3-030-05318-5_2.

[Var+21] Blesson Varghese et al. “A Survey on Edge Performance Benchmarking”. In: *ACM Computing Surveys* (Apr. 2021). ISSN: 0360-0300. DOI: 10.1145/3444692.

[Vas+17] Ashish Vaswani et al. “Attention is All you Need”. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*. 2017. URL: https://dl.acm.org/doi/10.5555/3295222.3295349.

[Vet+24] Anna Vettoruzzo et al. “Advances and Challenges in Meta-Learning: A Technical Review”. In: *Transactions on Pattern Analysis and Machine Intelligence* (2024), pp. 4763–4779. DOI: 10.1109/TPAMI.2024.3357847.

[Vin+20] Ricardo Vinuesa et al. “The Role of Artificial Intelligence in Achieving the Sustainable Development Goals”. In: *Nature Communications* (Jan. 2020), p. 233. ISSN: 2041-1723. DOI: 10.1038/s41467-019-14108-y.

[VLW25] Gael Varoquaux, Sasha Luccioni, and Meredith Whittaker. “Hype, Sustainability, and the Price of the Bigger-is-Better Paradigm in AI”. In: *Proceedings of the 8th Conference on Fairness, Accountability and Transparency (FAccT)*. 2025, pp. 61–75. ISBN: 9798400714825. DOI: 10.1145/3715275.3732006.

[Wan+19] Yuyang Wang et al. “Deep Factors for Forecasting”. In: *Proceedings of the 36th International Conference on Machine Learning (ICML)*. 2019, pp. 6607–6617. URL: https://proceedings.mlr.press/v97/wang19k.html.

[Web25] Web of Science. *Web of Science Document Search Results*. Accessed: 2025-03-26. 2025. URL: https://webofscience.com/wos/woscc/summary/812371c1-d876-4289-b0d0-6bf47b101aa3-01559870de/relevance/1.

[Wen+17] Ruofeng Wen et al. *A Multi-Horizon Quantile Recurrent Forecaster*. 2017. URL: https://arxiv.org/abs/1711.11053.

[Wil24] Wilo SE. *Sustainability Strategy and Programme*. Accessed: 2025-04-11. 2024. URL: https://wilo.com/en/Pioneering/Sustainability-strategy-and-programme/.

[Wis+24] Magdalena Wischnewski et al. "In Seal We Trust? Investigating the Effect of Certifications on Perceived Trustworthiness of AI Systems". In: *Human-Machine Communication* (2024), p. 7. DOI: 10.30658/hmc.8.7.

[WK25] Jeff Tollefson Witze and Max Kozlov. "What Trump 2.0 means for science: the likely winners and losers". In: *Nature* (2025), p. 533. DOI: 10.1038/d41586-025-00052-z.

[WKM23] Magdalena Wischnewski, Nicole Krämer, and Emmanuel Müller. "Measuring and Understanding Trust Calibrations for Automated Systems: A Survey of the State-Of-The-Art and Future Directions". In: *Proceedings of the 41st Conference on Human Factors in Computing Systems (CHI)*. 2023. ISBN: 978-1-4503-9421-5. DOI: 10.1145/3544548.3581197.

[WM97] David H. Wolpert and William G. Macready. "No Free Lunch Theorems for Optimization". In: *Transactions on Evolutionary Computation* (1997), pp. 67–82. DOI: 10.1109/4235.585893.

[Wol+20] Thomas Wolf et al. "Transformers: State-of-the-Art Natural Language Processing". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Oct. 2020, pp. 38–45. DOI: 10.18653/v1/2020.emnlp-demos.6.

[WRP19] Martin Wistuba, Ambrish Rawat, and Tejaswini Pedapati. *A Survey on Neural Architecture Search*. 2019. URL: https://arxiv.org/abs/1905.01392.

[WSS16] Martin Wistuba, Nicolas Schilling, and Lars Schmidt-Thieme. "Two-Stage Transfer Surrogate Model for Automatic Hyperparameter Optimization". In: *Machine Learning and Knowledge Discovery in Databases*. 2016, pp. 199–214. ISBN: 978-3-319-46128-1.

[Wu+22] Carole-Jean Wu et al. *Sustainable AI: Environmental Implications, Challenges and Opportunities*. 2022. URL: https://arxiv.org/abs/2111.00364.

[Wyn21] Aimee van Wynsberghe. "Sustainable AI: AI for Sustainability and the Sustainability of AI". In: *AI and Ethics* (Aug. 2021), pp. 213–218. ISSN: 2730-5961. DOI: 10.1007/s43681-021-00043-6.

[Yan14] Xin-She Yang. "Multi-Objective Optimization". In: *Nature-Inspired Optimization Algorithms*. Jan. 2014, pp. 197–211. ISBN: 978-0-12-416743-8. DOI: 10.1016/B978-0-12-416743-8.00014-2.

[Yeh07] Arthur B. Yeh. "A Modern Introduction to Probability and Statistics". In: *Technometrics* (2007), p. 359. DOI: 10.1198/TECH.2007.S502.

[Zho12] Zhi-Hua Zhou. *Ensemble Methods: Foundations and Algorithms*. 1st. Chapman & Hall/CRC, 2012. ISBN: 1439830037. URL: https://dl.acm.org/doi/10.5555/2381019.

[ZL17] Barret Zoph and Quoc Le. "Neural Architecture Search with Reinforcement Learning". In: *5th International Conference on Learning Representations (ICLR)*. 2017. URL: https://openreview.net/forum?id=r1Ue8Hcxg.

[ZLH21] Lucas Zimmer, Marius Lindauer, and Frank Hutter. "Auto-Pytorch: Multi-Fidelity MetaLearning for Efficient and Robust AutoDL". In: *Transactions on Pattern Analysis and Machine Intelligence* (2021), pp. 3079–3090. DOI: 10.1109/TPAMI.2021.3067763.

[Zop+18] Barret Zoph et al. "Learning Transferable Architectures for Scalable Image Recognition". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018, pp. 8697–8710. DOI: 10.1109/CVPR.2018.00907.

[ZSW22] Mengdi Zhang, Jun Sun, and Jingyi Wang. "Which Neural Network Makes More Explainable Decisions? An Approach Towards Measuring Explainability". In: *Automated Software Engineering* (Apr. 2022), p. 39. ISSN: 1573-7535. DOI: 10.1007/s10515-022-00338-w.

[Zub18] Shoshana Zuboff. *The Age of Surveillance Capitalism: The Fight for a Human Future at the New Frontier of Power*. 1st. PublicAffairs, 2018. ISBN: 1-61039-569-7. URL: https://www.publicaffairsbooks.com/titles/shoshana-zuboff/the-age-of-surveillance-capitalism/9781610395694/.