Applying AI in Practice: Key Challenges and Lessons Learned

Lukas Fischer, Lisa Ehrlinger, Verena Geist, Rudolf Ramler, Florian Sobieczky, Werner Zellinger, Bernhard Moser

Applying AI in Practice: Key Challenges and Lessons Learned*

Lukas Fischer[1 [0000−0001−5303−6638]], Lisa Ehrlinger[1,2 [0000−0001−5313−0368]], Verena Geist[1 [0000−0002−3729−1265]], Rudolf Ramler[1 [0000−0001−9903−6107]], Florian Sobieczky[1 [0000−0001−5228−0153]], Werner Zellinger[1 [0000−0003−1166−6062]], and Bernhard Moser[1 [0000−0003−1859−046X]]

[1] Software Competence Center Hagenberg GmbH (SCCH), Hagenberg, Austria
{lukas.fischer, lisa.ehrlinger, verena.geist, rudolf.ramler, florian.sobieczky, werner.zellinger, bernhard.moser}@scch.at
[2] Johannes Kepler University, Linz, Austria
lisa.ehrlinger@jku.at

Abstract. The main challenges along with lessons learned from ongoing research in the application of machine learning systems in practice are discussed, taking into account aspects of theoretical foundations, systems engineering, and human-centered AI postulates. The analysis outlines a fundamental theory-practice gap which superimposes the challenges of AI system engineering at the level of data quality assurance, model building, software engineering and deployment.

Keywords: Machine Learning Systems · Data Quality · Domain Adaptation · Hybrid Models · Software Engineering · Embedded Systems · Human Centered AI.

1 Introduction

Many real-world tasks are characterized by uncertainties and probabilistic data that is hard to understand and hard to process for humans. Machine learning and knowledge extraction [46] help turning this data into useful information for realizing a wide spectrum of applications such as image recognition, scene understanding, decision-support systems, etc. that enable new use cases across a broad range of domains.

The success of various machine learning methods, in particular Deep Neural Networks (DNNs), for challenging problems of computer vision and pattern recognition, has led to a "Cambrian explosion" in the field of Artificial Intelligence (AI). In many application areas, AI researchers have turned to deep

* Special thanks go to A Min Tjoa, former Scientific Director of SCCH, for his encouraging support in bringing together data and software science to tackle the research problems discussed in this paper. The research reported in this paper has been funded by BMK, BMDW, and the Province of Upper Austria in the frame of the COMET Programme managed by FFG.

learning as the solution of choice [54,97]. A characteristic of this development is the acceleration of progress in AI over the last decade, which has led to AI systems that are strong enough to raise serious ethical and societal acceptance questions. Another characteristic of this development is the way how such systems are engineered. Above all, there is an increasing interconnection of traditionally separate disciplines such as data analysis, model building and software engineering. In particular, data-driven AI methods such as DNNs allow data to shape models and software systems that operate them. System engineering of AI-driven software therefore faces novel challenges at all stages of the system lifecycle [51]:

- **Key Challenge 1:** *AI intrinsic challenges* due to peculiarities or shortcomings of today's AI methods; in particular, current data-driven AI is characterized by:
  - data challenge in terms of quality assurance and procurement;
  - challenge to integrate expert knowledge and models;
  - model integrity and reproducibility challenge due to unstable performance profiles triggered by small variations in the implementation or input data (adversarial noise);
- **Key Challenge 2:** Challenges in the process of *AI system engineering* ranging from requirements analysis and specification to deployment including
  - testing, debugging and documentation challenges;
  - challenge to consider the constraints of target platforms at design time;
  - certification and regulation challenges resulting from highly regulated target domains such as in a bio-medical laboratory setting;
- **Key Challenge 3:** *Interpretability and trust challenge* in the operational environment, in particular
  - trust challenge in terms of lack of interpretability and transparency by opaque models;
  - challenge posed by ethical guideline;
  - acceptance challenge in terms of societal barriers to AI adoption in society, healthcare or working environments;

2 Key Challenges on System Engineering Posed by Data-Driven AI

2.1 AI Intrinsic Challenges

There are peculiarities of deep learning methods that affect the correct interpretation of the system's output and the transparency of the system's configuration.

Lack of uniqueness of internal configuration: First of all, in contrast to traditional engineering, there is a lack of uniqueness of internal configuration causing difficulties in model comparison. Systems based on machine learning, in particular deep learning models, are typically regarded as black boxes. However, it is not just simply the complex nested non-linear structure which matters as often

pointed out in the literature, see [86]. There are mathematical or physical systems which are also complex, nested and non-linear, and yet interpretable (e.g., wavelets, statistical mechanics). It is an amazing, unexpected phenomenon that such deep networks become easier to be optimized (trained) with an increasing number of layers, hence complexity, see [110,100]. More precisely, to find a reasonable sub-optimum out of many equally good possibilities. As consequence, and in contrast to classical engineering, we lose uniqueness of the internal optimal state.

Lack of confidence measure: A further peculiarity of state of the art deep learning methods is the lack of confidence measure. In contrast to Bayesian based approaches to machine learning, most deep learning models do not offer a justified confidence measure of the model's uncertainties. E.g., in classification models, the probability vector obtained in the top layer (predominantly softmax output) is often interpreted as model confidence, see, e.g., [26] or [35]. However, functions like softmax can result in extrapolations with unjustified high confidence for points far from the training data, hence providing a false sense of safety [39]. Therefore, it seems natural to try to introduce the Bayesian approach also to DNN models. The resulting uncertainty measures (or, synonymously, confidence measures) rely on approximations of the posterior distribution regarding the weights given the data. As a promising approach in this context, variational techniques, e.g., based on Monte Carlo dropout [27], allow to turn these Bayesian concepts into computationally tractable algorithms. The variational approach relies on the Kullback-Leibler divergence for measuring the dissimilarity between distributions. As a consequence, the resultant approximating distribution becomes concentrated around a single mode, underestimating the uncertainty beyond this mode. Thus, the resulting measure of confidence for a given instance remains unsatisfactory and there might be still regions with misinterpreted high confidence.

Lack of control of high-dimensionality effects: Further, there is the still unsolved problem of lack of control of high-dimensionality effects. There are high dimensional effects which are not yet fully understood in the context of deep learning, see [31] and [28]. Such high-dimensional effects can cause instabilities as illustrated, for example, by the emergence of so-called adversarial examples, see e.g. [96,3].

2.2 AI System Engineering Challenges

In a data-driven AI systems there are two equally consequential components: software code and data. However, some input data are inherently volatile and may change over time. Therefore, it is important that these changes can be identified and tracked to fully understand the models and the final system. To this end, the development of such data-driven systems has all the challenges of traditional software engineering combined with specific machine learning problems causing additional hidden technical debts [87].

Theory-Practice Gap in Machine Learning: The design and test principles of machine learning are underpinned by statistical learning theory and its fundamental theorems such as Vapnik's theorem [99]. The theoretical analysis relies on idealized assumptions such as that the data are drawn independent and identically distributed from the same probability distribution. As outlined in [81], however, this assumption may be violated in typical applications such as natural language processing [48] and computer vision [106,108].

This problem of data set shifting can result from the way input characteristics are used, from the way training and test sets are selected, from data sparsity, from shifts in data distribution due to non-stationary environments, and also from changes in activation patterns within layers of deep neural networks. Such a data set shift can cause misleading parameter tuning when performing test strategies such as cross-validation [104,58].

This is why engineering machine learning systems largely relies on the skill of the data scientist to examine and resolve such problems.

Data Quality Challenge: While much of the research in machine learning and its theoretical foundation has focused on improving the accuracy and efficiency of training and inference algorithms, less attention has been paid to the equally important practical problem of monitoring the quality of the data supplied to machine learning [6,19]. Especially heterogeneous data sources, the occurrence of unexpected patterns, and a large number of schema-free data pose additional problems for data management which directly impact data extraction from multiple sources, data preparation, and data cleansing [7,84].

For data quality issues, the situation is similar to the detection of software bugs. The earlier the problems are detected and resolved, the better for model quality and development productivity.

Configuration Maintenance Challenge: ML system developers usually start from ready-made, pre-trained networks and try to optimize their execution on the target processing platform as much as possible. This practice is prone to the entanglement problem [87]: If changes are made to an input feature, the meaning, weighting, or use of the other features may also change. This means that machine learning systems must be designed so that feature engineering and selection changes are easily tracked. Especially when models are constantly revised and subtly changed, the tracking of configuration updates while maintaining the clarity and flexibility of the configuration become an additional burden.

Deployment Challenge: The design and training of the learning algorithm and the inference of the resulting model are two different activities. The training is very computationally intensive and is usually conducted on a high performance platform [103]. It is an iterative process that leads to the selection of an optimal algorithm configuration, usually known as hyperparameter optimization, with accuracy as the only major goal of the design [105]. While the training process is usually conducted offline, inference very often has to deal with real-time constraints, tight power or energy budgets, and security threats. This dichotomy

determines the need for multiple design re-spins (before a successful integration), potentially leading to long tuning phases, overloading the designers and producing results highly depending on their skills. Despite the variety of resources available, optimizing these heterogeneous computing architectures for performing low-latency and energy-efficient DL inference tasks without compromising performance is still a challenge [5].

2.3 Interpretability and Trust Challenge

In contrast to traditional computing, AI can now perform tasks that previously only humans were able to do. As such it contains the possibility to revolutionize every aspect of our society. The impact is far-reaching. First, with the increasing spread of AI systems, the interaction between humans and AI will increasingly become the dominant form of human-computer interaction [1]. Second, this development will shape the future workforce. PwC[3] predicts a relatively low displacement of jobs (around 3%) in the first wave of AI, but this could dramatically increase up to 30% by the mid-2030's. Therefore, human centered AI has started coming to the forefront of AI research based on postulated ethical principles for protecting human autonomy and preventing harm. Recent initiatives at national[4] and supra-national[5] level emphasize the need for research in trusted AI.

Interpretability Challenge: Essential aspects of trusted AI are explainability and interpretability. While interpretability is about being able to discern the mechanics without necessarily knowing why. Explainability is being able to quite literally explain what is happening, for example, by referring to mechanical laws. It is well known that the great successes of machine learning in recent decades in terms of applicability and acceptance are relativized by the fact that they can be explained less easily with increasing complexity of the learning model [60,44,90]. Explainability of the solution is thus increasingly perceived as an inherent quality of the respective methods [90,9,15,33]. Particularly in the case of deep learning methods attempts to interpret the predictions made using parameters fail [33]. The necessity to obtain not only increasing prediction accuracy but also the interpretation of the solutions determined by ML or Deep Learning arises at the latest with the ethical [76,10], legal [13], psychological [59], medical [25,45], and sociological [111] questions tied to their application. The common element of these questions is the demand to clearly interpret the decisions proposed by artificial intelligence (AI). The complex of problems that derives from this aspect of artificial intelligence for explainability, transparency, trustworthiness, etc. is generally described with the term Explainable Artificial Intelligence,

[3] https://www.pwc.com/gx/en/services/people-organisation/workforce-of-the-future/workforce-of-the-future-the-competing-forces-shaping-2030-pwc.pdf

[4] https://www.whitehouse.gov/wp-content/uploads/2019/06/National-AI-Research-and-Development-Strategic-Plan-2019-Update-June-2019.pdf

[5] https://ec.europa.eu/digital-single-market/en/news/ethics-guidelines-trustworthy-ai

synonymously "Explainable AI" or "XAI". Its broad relevance can be seen in the interdisciplinary nature of the scientific discussion that is currently taking place on such terms as interpretation, explanation and refined versions such as causability and causality in connection with AI methods [33,30,42,43].

Trust Challenge: In contrast to Interpretability, trust is a much more comprehensive concept. Trust is linked to the uncertainty about a possible malfunctioning or failure of the AI system as well as to circumstances of delegating control to a machine as a "black box". Predictability and dependability of AI technology as well as the understanding of the technology's operations and the intentions of its creators are essential drivers of trust [12]. Particularly, in critical applications the user wants to understand the rationale behind a classification, and under which conditions the system is trustful and when not. Consequently, AI systems must make it possible to take these human needs of trust and social compatibility into account. On the other hand, we have to be aware of limitations and peculiarities of state of the art AI systems. Currently, the topic of trusted AI is discussed in different communities at different levels of abstraction:

- in terms of high level ethical guidelines (e.g. ethics boards such as algorithmwatch.org[6], EU's Draft Ethics Guidelines[7]);
- in terms of regulatory postulates for current AI systems regarding e.g. transparency (working groups on standardization, e.g. ISO/IEC JTC 1/SC 42 on artificial intelligence[8]);
- in terms of improved features of AI models (above all by explainable AI community [34,41]);
- in terms of trust modeling approaches (e.g. multi-agent systems community [12]).

In view of the model-intrinsic and system-technical challenges of AI that have been pointed out in the sections 2.1 and 2.2, the gap between the envisioned high-level ethical guidelines of human-centered AI and the state of the art of AI systems becomes evident.

3 Approaches, In-Progress Research and Lessons Learned

In this section we discuss ongoing research facing the outlined challenges in the previous section, comprising:

(1) Automated and Continuous Data Quality Assurance, see section 3.1;
(2) Domain Adaptation Approach for Tackling Deviating Data Characteristics at Training and Test Time, see section 3.2;
(3) Hybrid Model Design for Improving Model Accuracy, see section 3.3;

[6] https://algorithmwatch.org/en/project/ai-ethics-guidelines-global-inventory/
[7] https://ec.europa.eu/digital-single-market/en/news/ethics-guidelines-trustworthy-ai
[8] https://www.iso.org/committee/6794475/x/catalogue/p/0/u/1/w/0/d/0

(4) Interpretability by Correction Model Approach, see section 3.4;
(5) Software Quality by Automated Code Analysis and Documentation Generation, see section 3.5;
(6) The ALOHA Toolchain for Embedded Platforms, see section 3.6;
(7) Human AI Teaming as Key to Human Centered AI, see section 3.7.

3.1 Approach 1: Automated and Continuous Data Quality Assurance

In times of large and volatile amounts of data, which are often generated automatically by sensors (e.g., in smart home solutions of housing units or industrial settings), it is especially important to, (i), automatically, and, (ii), continuously monitor the quality of data [22,88]. A recent study [20] shows that the continuous monitoring of data quality is only supported by very few software tools. In the open-source area these are Apache Griffin[9], MobyDQ[10], and QuaIIe [21]. Apache Griffin and QuaIIe implement data quality metrics from the reference literature (see [21,40]), whereby most of them require a reference database (gold standard) for calculation. MobyDQ, on the other hand, is rule-based, with the focus on data quality checks along a pipeline, where data is compared between two different databases. Since existing open-source tools were insufficient for the permanent measurement of data quality within a database or a data stream used for data analysis and machine learning, we developed the Data Quality Library (DaQL). DaQL allows the extensive definition of data quality rules , based on the newly developed DaQL language. These rules do not require reference data and DaQL has already been used for a ML application in an industrial setting [19]. However, to ensure their validity, the rules for DaQL are created manually by domain experts.

Lesson Learned: In literature, data quality is typically defined with the "fitness for use" principle, which illustrates the high contextual dependency of the topic [11,102]. Thus, one important lesson learned is the need for more research into the automated generation of domain-specific data quality rules. In addition, the integration of contextual knowledge (e.g., the respective ML model using the data) needs to be considered. Here, knowledge graphs pose a promising solution, which indicates that knowledge about the quality of data is part of the bigger picture outlined in Approach (and lesson learned) 7: the usage of knowledge graphs to interpret the quality of AI systems. In addition to the measurement (i.e., detection) of data quality issues, we consider research into the automated correction (i.e., cleansing) of sensor data as additional challenge [18]. Especially since automated data cleansing poses the risk to insert new errors in the data (cf. [63]), which is specifically critical in enterprise settings.

[9] https://griffin.incubator.apache.org
[10] https://github.com/mobydq/mobydq

3.2 Approach 2: The Domain Adaptation Approach for Tackling Deviating Data Characteristics at Training and Test Time

In [106] and [108] we introduce a novel distance measure, the so-called Centralized Moment Discrepancy (CMD), for aligning probability distributions in the context of domain adaption. Domain adaptation algorithms are designed to minimize the misclassification risk of a discriminative model for a target domain with little training data by adapting a model from a source domain with a large amount of training data. Standard approaches measure the adaptation discrepancy based on distance measures between the empirical probability distributions in the source and target domain, i.e., in our setting this means training time and test time, respectively. In [109] we can show that our CMD approach, refined by practice-oriented information-theoretic assumptions of the involved distributions, yield a generalization of the conditions of Vapnik's theorem [99].

As a result we obtain quantitative generalization bounds for recently proposed moment-based algorithms for unsupervised domain adaptation which perform particularly well in many practical tasks [95,106,108,74,107].

Lesson Learned: It is interesting that moment-based probability distance measure are the most weakest among those utilized in the machine learning and, in particular, domain adaptation. Weak in this setting means that convergence by the stronger distance measures entails convergence of the weaker. Our lesson learned is that a weaker distance measure can be more robust than stronger distance measures. At the first glance, this observation might appear counter-intuitive. However, at a second look, it becomes intuitive that the minimization of stronger distance measures are more prone to the effect of negative transfer [77], i.e. the adaptation of source-specific information not present in the target domain. Further evidence can be found in the area of generative adversarial networks where the alignment of distributions by strong probability metrics can cause problems of mode collapse which can be mitigated by choosing weaker similarity concepts [17]. Thus, it is better to abandon stronger concepts of similarity in favour of weaker ones and to use stronger concepts only if they can be justified.

3.3 Approach 3: Hybrid Model Design for Improving Model Accuracy by Integrating Expert Hints in Biomedical Diagnostics

For diagnostics based on biomedical image analysis, image segmentation serves as a prerequisite step to extract quantitative information [70]. If, however, segmentation results are not accurate, quantitative analysis can lead to results that misrepresent the underlying biological conditions [50]. To extract features from biomedical images at a single cell level, robust automated segmentation algorithms have to be applied. In the Austrian FFG project VISIOMICS[11], which

[11] Platform supporting an integrated analysis of image and multiOMICs data based on liquid biopsies for tumor diagnostics – https://www.visiomics.at/

is devoted to cell analysis, we tackle this problem by following a cell segmentation ensemble approach, consisting of several state-of-the-art deep neural networks [85,38]. In addition to overcome the lack of training data, which is very time consuming to prepare and annotate, we utilize a Generative Adversarial Network approach (GANs) for artificial training data generation [53][12]. The underlying dataset was also published [52] and is available online[13]. Particularly for cancer diagnostics, clinical decision-making often relies on timely and cost-effective genome-wide testing. Similar to biomedical imaging, classical bioinformatic algorithms, often require manual data curation, which is error prone, extremely time-consuming, and thus has negative effects on time and cost efficiency. To overcome this problem, we developed the DeepSNP[14] network to learn from genome-wide single-nucleotide polymorphism array (SNPa) data and to classify the presence or absence of genomic breakpoints within large genomic windows with high precision and recall [16].

Lesson Learned: First, it is crucial to rely on expert knowledge when it comes to data augmentation strategies. This becomes more important the more complex the data is (high number of cores and overlapping cores). Less complex images do not necessarily benefit from data augmentation. Second, by introducing so-called localization units the network is able to gain the ability to exactly localize anomalies in terms of genomic breakpoints despite never experiencing their exact location during training. In this way we have learned that localization and attention units can be used to significantly ease the effort of annotating data.

3.4 Approach 4: Interpretability by Correction Model Approach

Last year, at a symposium on predictive analytics in Vienna [93], we introduced an approach to the problem of formulating interpretability of AI models for classification or regression problems [37] with a given basis model, e.g., in the context of model predictive control [32]. The basic idea is to root the problem of interpretability in the basic model by considering the contribution of the AI model as correction of this basis model and is referred to as "Before and After Correction Parameter Comparison (BAPC)". The idea of small correction is a common approach in mathematics in the field of perturbation theory, for example of linear operators. In [91,92] the idea of small-scale perturbation (in the sense of linear algebra) was used to give estimates of the probability of return of an odyssey on a percolation cluster. The notion of "small influence" appears here in a similar way via the measures of determination for the AI model compared to the basic model.

According to BAPC, an AI-based correction of a solution of these problems, which is previously provided by a basic model, is interpretable in the sense of

[12] Nuclear Segmentation Pipeline code available: https://github.com/SCCH-KVS/NuclearSegmentationPipeline
[13] BioStudies: https://www.ebi.ac.uk/biostudies/studies/S-BSST265
[14] DeepSNP code available: https://github.com/SCCH-KVS/deepsnp

this basic model, if its effect can be described by its parameters. Since this effect refers to the estimated target variables of the data. In other words, an AI correction in the sense of a basic model is interpretable in the sense of this basic model exactly when the accompanying change of the target variable estimation can be characterized with the solution of the basic model under the corresponding parameter changes. The basic idea of the approach is thus to apply the explanatory power of the basic model to the correcting AI method in that their effect can be formulated with the help of the parameters of the basic model. BAPC's ability to use the basic model to predict the modified target variables makes it a so-called surrogate [9].

The proposed solution for the interpretation of the AI correction is of course limited from the outset by the interpretation horizon of the basic model. Furthermore, it must be assumed that the basic model is too weak to describe the phenomena underlying the correction in accordance with the actual facts. We therefore distinguish between explainability and interpretability and, with the definition of interpretability in terms of the basic model introduced above, we do not claim to always be able to explain, but rather to be able to describe (i.e. interpret) the correction as a change of the solution using the basic model. This is achieved by means of the features used in the basic model and their modified parameters. As with most XAI approaches (e.g., feature importance vector [33]), the goal is to find the most significant changes in these parameters.

Lesson Learned: This approach is work in progress and will be tackled in detail in the upcoming Austrian FFG research project "inAIco". As lesson learned we appreciate the BAPC approach as result of interdisciplinary research at the intersection of mathematics, machine learning and model predictive control. We expect that the approach generally only works for "small" AI corrections. It must be possible to formulate conditions about the size (i.e. "smallness") of the AI correction under which the approach will work in any case. However, it is an advantage of our approach that interpretability does not depend on human understanding (see the discussion in [33] and [9]). An important aspect is its mathematical rigidity, which avoids the accusation of "quasi-scientificity" (see [57]).

3.5 Approach 5: Software Quality by Code Analysis and Automated Documentation

Quality assurance measures in software engineering include, e.g., automated testing [2], static code analysis [73], system redocumentation [69], or symbolic execution [4]. These measures need to be risk-based [23,83], exploiting knowledge about system and design dependencies, business requirements, or characteristics of the applied development process.

AI-based methods can be applied to extract knowledge from source code or test specifications to support this analysis. In contrast to manual approaches, which require extensive human annotation work, machine learning methods have

been applied for various extraction and classification tasks, such as comment classification of software systems with promising results in [78,89,94].

Software engineering approaches contribute to automate (i) AI-based system testing, e.g., by means of predicting fault-prone parts of the software system that need particular attention [68], and (ii) system documentation to improve software maintainability [98,69,14] and to support re-engineering and migration activities [14]. In particular, we developed a feed-back directed testing approach to derive tests from interacting with a running system [61], which we successfully applied in various industry projects [82,24]. In an ongoing redocumentation project [29], we automatically generate parts of the functional documentation, containing business rules and domain concepts, and all the technical documentation.

Lesson Learned: Keeping documentation up to date is essential for the maintainability of frequently updated software and to minimise the risk of technical debt due to the entanglement of data and sub-components of machine learning systems. The lesson learned is that for this problem also machine learning can be utilized when it comes to establishing rules for detecting and classifying comments (accuracy of $> 95\%$) and integrating them when generating readable documentation.

3.6 Approach 6: The ALOHA Toolchain for Embedded Platforms

In [66] and [65] we introduce ALOHA, an integrated tool flow that tries to make the design of deep learning (DL) applications and their porting on embedded heterogeneous architectures as simple and painless as possible. ALOHA is the result of interdisciplinary research funded by the EU[15]. The proposed tool flow aims at automating different design steps and reducing development costs by bridging the gap between DL algorithm training and inference phases. The tool considers hardware-related variables and security, power efficiency, and adaptivity aspects during the whole development process, from pre-training hyperparameter optimization and algorithm configuration to deployment. According to Figure 1 the general architecture of the ALOHA software framework [67] consists of three major steps:

- (Step 1) algorithm selection,
- (Step 2) application partitioning and mapping, and
- (Step 3) deployment on target hardware.

Starting from a user-specified set of input definitions and data, including a description of the target architecture, the tool flow generates a partitioned and mapped neural network configuration, ready to the target processing architecture, which also optimizes predefined optimization criteria. The criteria for optimization include both application-level accuracy and the required security level,

[15] https://www.aloha-h2020.eu/

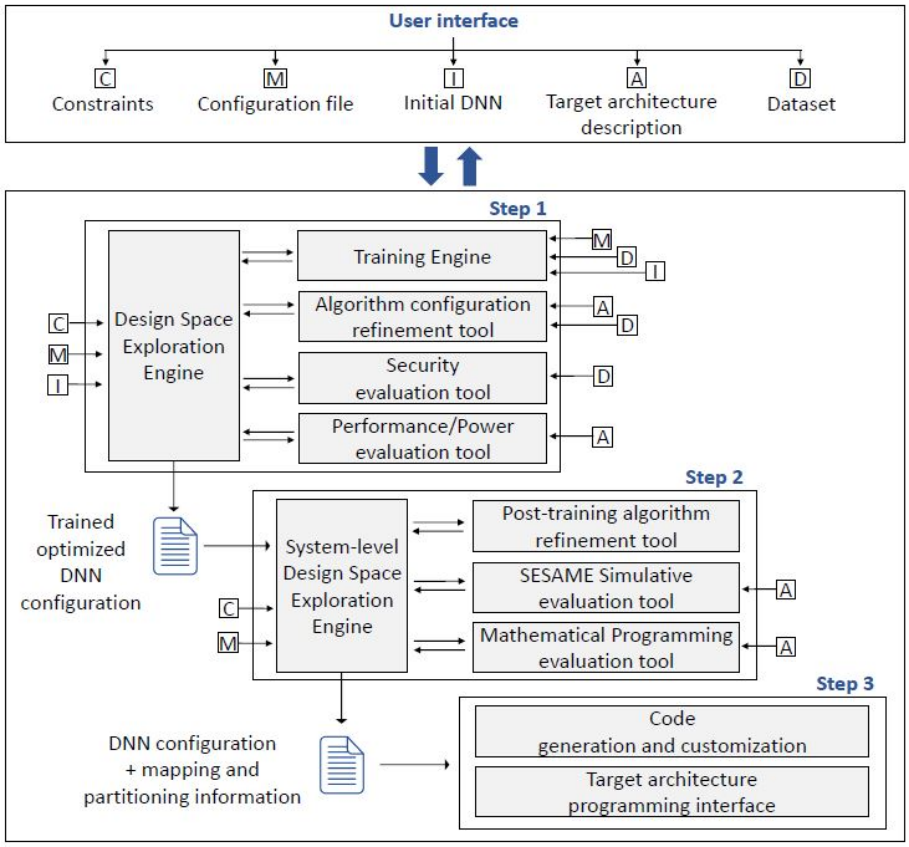

Fig. 1. General architecture of the ALOHA software framework. Nodes in the upper part of the figure represent the key inputs of the tool flow specified by the users, for details see [67].

Inference execution time and power consumption. A RESTful microservices approach allows each step of the development process to be broken down into smaller, completely independent components that interact and influence each other through the exchange of HTTP calls [71]. The implementations of the various components are managed using a container orchestration platform. The standard ONNX[16] (Open Neural Network Exchange) is used to exchange deep learning models between the different components of the tool flow.

In Step 1 a Design Space comprising admissible model architectures for hyperparamerter tuning is defined. This Design Space is configured via satellite tools that evaluate the fitness in terms of the predefined optimization criteria such as accuracy (by the Training Engine), robustness against adversarial attacks

[16] https://onnx.ai/

(by the Security evaluation tool) and power (by the Power evaluation tool). The optimization is based on a) hyperparameter tuning based on a non-stochastic infinite-armed bandit approach [55], and b) a parsimonious inference strategy that aims to reduce the bit depth of the activation values from initially 8bit to 4bit by a iterative quantization and retraining steps [47]. The optimization in Step 2 exploits genetic algorithm for surfing the design space and requiring evaluation of the candidate partitioning and mapping scheme to the satellite tools Sesame [80] and Architecture Optimization Workbench (AOW) [62].

The gain in performance was evaluated in terms of inference time needed to execute the modified model on NEURAghe [64], a Zynq-based processing platform that contains both a dual ARM Cortex A9 processor (667 MHz) and a CNN accelerator implemented in the programmable logic. The statistical analysis on the switching activity of our reference models showed that, on average, only about 65% of the kernels are active in the layers of the network throughout the target validation data set. The resulting model loses only 2% accuracy (baseline 70%) while achieving an impressive 48.31% reduction in terms of FLOPs.

Lesson Learned: Following the standard training procedure deep models tend to be oversized. This research shows that some of the CNN layers are operating in a static or close-to-static mode, enabling the permanent pruning of the redundant kernels from the model. But, the second optimization strategy dedicated to parsimonious inference turns out to more effective on pure software execution, since it more directly deactivates operations in the convolution process. All in all, this study shows that there is a lot of potential for optimisation and improvement compared to standard deep learning engineering approaches.

3.7 Approach 7: Human AI Teaming Approach as Key to Human Centered AI

In [36], we introduce an approach for human-centered AI in working environments utilizing knowledge graphs and relational machine learning ([72,79]). This approach is currently being refined in the ongoing Austrian project Human-centred AI in digitised working environments (AI@Work). The discussion starts with a critical analysis of the limitations of current AI systems whose learning/training is restricted to predefined structured data, most vector-based with a pre-defined format. Therefore, we need an approach that overcomes this restriction by utilizing a relational structures by means of a knowledge graph (KG) that allows to represent relevant context data for linking ongoing AI-based and human-based actions on the one hand and process knowledge and policies on the other hand. Figure 2 outlines this general approach where the knowledge graph is used as an intermediate representation of linked data to be exploited for improvement of the machine learning system, respectively AI system. Methods applied in this context will include knowledge graph completion techniques that aim at filling missing facts within a knowledge graph [75]. The KG flexibly will allow tying together contextual knowledge about the team of involved human and AI based actors including interdependence relations, skills and tasks

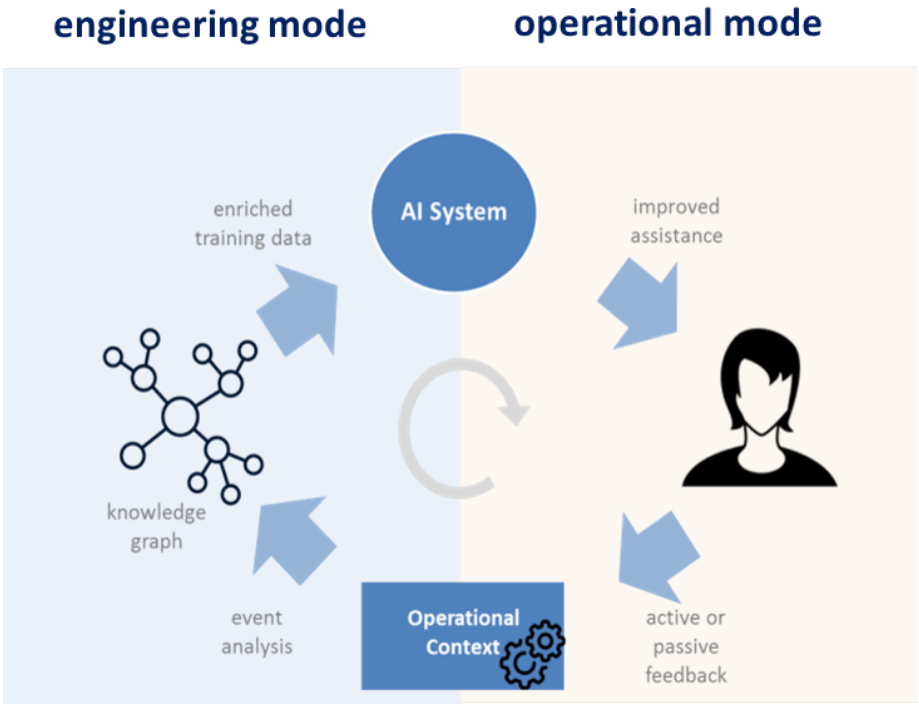

Fig. 2. A knowledge-graph approach to enhance vector-based machine learning in order to support human AI teaming by taking context and process knowledge into account.

together with application and system process and organizational knowledge [49]. Relational machine learning will be developed in combination with an updatable knowledge graph embedding [8,101]. This relational ML will be exploited for analysing and mining the knowledge graph for the purpose of detecting inconsistencies, curating, refinement, providing recommendations for improvements and detecting compliance conflicts with predefined behavioural policies (e.g. ethic or safety policies). The system will learn from the environment, user feedback, changes in the application or deviations from committed behavioral patterns in order to react by providing updated recommendations or triggering actions in case of compliance conflicts. But, the construction of the knowledge graph and keeping it up-to-date is a critical step as it usually includes laborious efforts for knowledge extraction, knowledge fusion, knowledge verification and knowledge updates. In order to address this challenge, our approach pursues bootstrapping strategies for knowledge extraction by recent advances in deep learning and embedding representations as promising methods for matching knowledge items represented in diverse formats.

Lesson Learned: As pointed out in Section 2.3 there is a substantial gap between current state-of-the-art research of AI systems and the requirements posed by ethical guidelines. Future research will rely much more on machine learning on graph structures. Fast updatable knowledge graphs and related knowledge graph embeddings might a key towards ethics by design enabling human centered AI.

4 Discussion and Conclusion

This paper can only give a small grasp of the broad field of AI research in connection with the application of AI in practice. The associated research is indeed inter- and even transdisciplinary [56]. Whatever, we come to the conclusion that a discussion on "Applying AI in Practice" needs to start with its theoretical foundations and a critical discussion about the limitations of current data-driven AI systems as outlined in Section 2.1. Approach 1, Section 3.1, and Approach 2, Section 3.2, help to stick to the theoretical prerequisites. Approach 1 contributes by reducing errors in the data and Approach 2 by extending the theory by relaxing its preconditions, bringing statistical learning theory closer to the needs of practice. However, building such systems and addressing the related challenges as outlined in Section 2.2 requires a bunch of skills from different fields, predominantly model building and software engineering know-how. Approach 3, Section 3.3, and Approach 4, Section 3.4, contribute to model building: Approach 3 by creatively adopting novel hybrid machine learning model architectures and Approach 4 by means of system theory that investigates AI as addendum to a basis model in order to be able to establish a notion of interpretability in a strict mathematical sense. Every model applied in practice must be coded in software. Approach 5, Section 3.5, outlines helpful state-of-the-art approaches in software engineering for maintaining the engineered software in good traceable and reusable quality which becomes more and more important with increasing complexity. Approach 6, Section 3.6, is an integrative approach that takes all the aspects discussed so far into account by proposing a software framework that supports the developer in all these steps when optimizing an AI system for an embedded platform. Finally, the challenge for human centered AI as outlined in Section 2.3 is somehow beyond of the state of the art. While the Key Challenges 1 and 2 require, above all, progress in the respective disciplines, Key Challenge 3 addressing "trust" in the end will require a mathematical theory of trust, that is a trust modeling approach at the level of system engineering that takes the psychological and cognitive aspects of human trust into account as well. Approach 7, Section 3.7, contributes to this endeavour by its conceptional approach for human AI teaming and its analysis of its prerequisites from relational machine learning.

References

1. Amershi, S., Weld, D., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P., Suh, J., Iqbal, S., Bennett, P.N., Inkpen, K., Teevan, J., Kikin-Gil, R., Horvitz, E.: Guidelines for Human-AI Interaction. In: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems. CHI '19 (2019)
2. Anand, S., Burke, E.K., Chen, T.Y., Clark, J., Cohen, M.B., Grieskamp, W., Harman, M., Harrold, M.J., Mcminn, P., Bertolino, A., et al.: An orchestrated survey of methodologies for automated software test case generation. Journal of Systems and Software **86**(8), 1978–2001 (2013)
3. Athalye, A., Engstrom, L., Ilyas, A., Kwok, K.: Synthesizing robust adversarial examples. arXiv e-prints (2017)
4. Baldoni, R., Coppa, E., D'elia, D.C., Demetrescu, C., Finocchi, I.: A survey of symbolic execution techniques. ACM Computing Surveys (CSUR) **51**(3), 1–39 (2018)
5. Bensalem, M., Dizdarević, J., Jukan, A.: Modeling of Deep Neural Network (DNN) Placement and Inference in Edge Computing. arXiv e-prints (2020)
6. Breck, E., Zinkevich, M., Polyzotis, N., Whang, S., Roy, S.: Data validation for machine learning. In: Proceedings of SysML (2019)
7. Cagala, T.: Improving data quality and closing data gaps with machine learning. In: Settlements, B.f.I. (ed.) Data needs and Statistics compilation for macroprudential analysis, vol. 46 (2017)
8. Cai, H., Zheng, V.W., Chang, K.C.C.: A comprehensive survey of graph embedding: Problems, techniques and applications (2017)
9. Carvalho, D.V., Pereira, E.M., Cardoso, J.S.: Machine learning interpretability: A survey on methods and metrics. Electronics **8**(8), 832 (2019)
10. Char, D.S., Shah, N.H., Magnus, D.: Implementing machine learning in health care — addressing ethical challenges. New England Journal of Medicine **378**(11), 981–983 (2018). https://doi.org/10.1056/NEJMp1714229, pMID: 29539284
11. Chrisman, N.: The role of quality information in the long-term functioning of a Geographic Information System. Cartographica The International Journal for Geographic Information and Geovisualization **21**(2), 79–88 (1983)
12. Cohen, R., Schaekermann, M., Liu, S., Cormier, M.: Trusted AI and the contribution of trust modeling in multiagent systems. In: Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS '19, 2019. pp. 1644–1648 (2019)
13. Deeks, A.: The judicial demand for explainable artificial intelligence. Columbia Law Review **119**(7), 1829–1850 (2019)
14. Dorninger, B., Moser, M., Pichler, J.: Multi-language re-documentation to support a cobol to java migration project. In: 2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER). pp. 536–540. IEEE (2017)
15. Doshi-Velez, F., Kim, B.: Towards a rigorous science of interpretable machine learning. arXiv (2017)
16. Eghbal-Zadeh, H., Fischer, L., Popitsch, N., Kromp, F., Taschner-Mandl, S., Gerber, T., Bozsaky, E., Ambros, P.F., Ambros, I.M., Widmer, G., Moser, B.A.: DeepSNP: An End-to-End Deep Neural Network with Attention-Based Localization for Breakpoint Detection in Single-Nucleotide Polymorphism Array Genomic Data. Journal of Computational Biology **26**(6) (2018)

17. Eghbal-zadeh, H., Zellinger, W., Widmer, G.: Mixture density generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5820–5829 (2019)
18. Ehrlinger, L., Grubinger, T., Varga, B., Pichler, M., Natschläger, T., Zeindl, J.: Treating Missing Data in Industrial Data Analytics. In: 2018 Thirteenth International Conference on Digital Information Management (ICDIM). pp. 148–155. IEEE (September 2018)
19. Ehrlinger, L., Haunschmid, V., Palazzini, D., Lettner, C.: A DaQL to Monitor Data Quality in Machine Learning Applications. In: Proceedings of the International Conference on Database and Expert Systems Applications (DEXA). Lecture Notes in Computer Science, vol. 11706, pp. 227–237. Springer, Cham, Switzerland (August 2019)
20. Ehrlinger, L., Rusz, E., Wöß, W.: A Survey of Data Quality Measurement and Monitoring Tools. CoRR **abs/1907.08138** (2019)
21. Ehrlinger, L., Werth, B., Wöß, W.: Automated Continuous Data Quality Measurement with QuaIIe. International Journal on Advances in Software **11**(3&4), 400–417 (December 2018)
22. Ehrlinger, L., Wöß, W.: Automated Data Quality Monitoring. In: 22nd MIT International Conference on Information Quality (ICIQ 2017). pp. 15.1–15.9 (2017)
23. Felderer, M., Ramler, R.: Integrating risk-based testing in industrial test processes. Software Quality Journal **22**(3), 543–575 (2014)
24. Fischer, S., Ramler, R., Linsbauer, L., Egyed, A.: Automating test reuse for highly configurable software. In: Proceedings of the 23rd International Systems and Software Product Line Conference-Volume A. pp. 1–11 (2019)
25. Forcier, M.B., Gallois, H., Mullan, S., Joly, Y.: Integrating artificial intelligence into health care through data access: can the gdpr act as a beacon for policymakers? Journal of Law and the Biosciences **6**(1), 317–335 (2019)
26. Gal, Y.: Uncertainty in deep learning. In: Thesis (2016)
27. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48. p. 1050–1059. ICML'16, JMLR.org (2016)
28. Galloway, A., Taylor, G.W., Moussa, M.: Predicting adversarial examples with high confidence. arXiv e-prints (2018)
29. Geist, V., Moser, M., Pichler, J., Beyer, S., Pinzger, M.: Leveraging machine learning for software redocumentation. In: 2020 IEEE 27th International Conference on Software Analysis, Evolution and Reengineering (SANER). pp. 622–626. IEEE (2020)
30. Gilpin, L.H., Bau, D., Yuan, B.Z., Bajwa, A., Specter, M., Kagal, L.: Explaining explanations: An overview of interpretability of machine learning. In: 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA). pp. 80–89 (2018)
31. Gorban, A.N., Tyukin, I.Y.: Blessing of dimensionality: mathematical foundations of the statistical physics of data. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **376**(2118) (2018)
32. Grancharova, A., Johansen, T.A.: Nonlinear Model Predictive Control, pp. 39–69. Springer Berlin Heidelberg, Berlin, Heidelberg (2012)
33. Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., Pedreschi, D.: A survey of methods for explaining black box models. ACM Comput. Surv. **51**(5) (2018)

34. Gunning, D.: Darpa's explainable artificial intelligence (XAI) program. In: Proceedings of the 24th International Conference on Intelligent User Interfaces. p. ii. IUI '19, Association for Computing Machinery, New York, NY, USA (2019)
35. Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q.: On calibration of modern neural networks. arXiv e-prints (2017)
36. Gusenleitner, N., Siedl, S., Stübl, G., Polleres, A., Recski, G., Sommer, R., Leva, M.C., Pichler, M., Kopetzky, T., Moser, B.A.: Facing mental workload in ai-transformed working environments (2019), h-WORKLOAD 2019: 3rd International Symposium on Human Mental Workload: Models and Applications
37. Hastie, T., Tibshirani, R., Friedman, J.: The elements of statistical learning: data mining, inference and prediction. Springer, 2 edn. (2009)
38. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: IEEE International Conference on Computer Vision (ICCV) (2017), arXiv: 1703.06870
39. Hein, M., Andriushchenko, M., Bitterwolf, J.: Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 41–50 (2019)
40. Heinrich, B., Hristova, D., Klier, M., Schiller, A., Szubartowicz, M.: Requirements for Data Quality Metrics. Journal of Data and Information Quality **9**(2) (2018)
41. Hoffman, R.R., Mueller, S.T., Klein, G., Litman, J.: Metrics for explainable AI: challenges and prospects. CoRR **abs/1812.04608** (2018)
42. Holzinger, A.: Interactive machine learning for health informatics: When do we need the human-in-the-loop? Brain Informatics **3**(2), 119–131 (5 2016). https://doi.org/10.1007/s40708-016-0042-6
43. Holzinger, A., Carrington, A., Müller, H.: Measuring the quality of explanations: The system causability scale (scs). In: Special Issue on Interactive Machine Learning. Künstliche Intelligenz (German Journal of Artificial intelligence), vol. 34, pp. 193—-198. TU Darmstadt (2020)
44. Holzinger, A., Kieseberg, P., Weippl, E., Tjoa, A.: Current advances, trends and challenges of machine learning and knowledge extraction: From machine learning to explainable ai. In: Machine Learning and Knowledge Extraction. CD-MAKE 2018. pp. 1–8. Lecture Notes in Computer Science, Springer International (2018)
45. Holzinger, A., Langs, G., Denk, H., Zatloukal, K., Müller, H.: Causability and explainability of artificial intelligence in medicine. WIREs Data Mining and Knowledge Discovery **9**(4), e1312 (2019)
46. Holzinger, A.: Introduction to machine learning and knowledge extraction (make). Mach. Learn. Knowl. Extr **1**(1), 1–20 (2017)
47. Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A.G., Adam, H., Kalenichenko, D.: Quantization and training of neural networks for efficient integer-arithmetic-only inference. CoRR **abs/1712.05877** (2017)
48. Jiang, J., Zhai, C.: Instance weighting for domain adaptation in NLP. In: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics. pp. 264–271 (2007)
49. Johnson, M., Vera, A.: No ai is an island: The case for teaming intelligence. AI Magazine **40**(1), 16–28 (2019)
50. Jung, C., Kim, C.: Impact of the accuracy of automatic segmentation of cell nuclei clusters on classification of thyroid follicular lesions. Cytometry. Part A: The Journal of the International Society for Analytical Cytology **85**(8), 709–718 (2014)

51. Kelly, C.J., Karthikesalingam, A., Suleyman, M., Corrado, G., King, D.: Key challenges for delivering clinical impact with artificial intelligence. BMC Medicine **17**(1), 195 (2019)
52. Kromp, F., Bozsaky, E., Rifatbegovic, F., Fischer, L., Ambros, M., Berneder, M., Weiss, T., Lazic, D., Dörr, W., Hanbury, A., Beiske, K., Ambros, P.F., Ambros, I.M., Taschner-Mandl, S.: An annotated fluorescence image dataset for training nuclear segmentation methods. Nature Scientific Data (2020), in press
53. Kromp, F., Fischer, L., Bozsaky, E., Ambros, I., Doerr, W., Taschner-Mandl, S., Ambros, P., Hanbury, A.: Deep learning architectures for generalized immunofluorescence based nuclear image segmentation. arXiv e-prints (2019)
54. Lecun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(FEB), 436–444 (2015)
55. Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., Talwalkar, A.: Hyperband: A novel bandit-based approach to hyperparameter optimization. J. Mach. Learn. Res. **18**(1), 6765–6816 (2017)
56. Li, S., Wang, Y.: Research on interdisciplinary characteristics: A case study in the field of artificial intelligence. IOP Conference Series: Materials Science and Engineering **677**, 052023 (2019)
57. Lipton, Z.C.: The mythos of model interpretability. Queue **16**(3), 31—-57 (2018)
58. Little, M.A., Varoquaux, G., Saeb, S., Lonini, L., Jayaraman, A., Mohr, D.C., Kording, K.P.: Using and understanding cross-validation strategies. perspectives on saeb et al. GigaScience **6**(5) (2017)
59. Lombrozo, T.: Explanatory preferences shape learning and inference. Trends in Cognitive Sciences **20**(10), 748–759 (2016)
60. London, A.: Artificial intelligence and black-box medical decisions: Accuracy versus explainability. The Hastings Center Report **49**, 15–21 (2019)
61. Ma, L., Artho, C., Zhang, C., Sato, H., Gmeiner, J., Ramler, R.: Grt: Program-analysis-guided random testing (t). In: 2015 30th IEEE/ACM International Conference on Automated Software Engineering (ASE). pp. 212–223. IEEE (2015)
62. Masin, M., Limonad, L., Sela, A., Boaz, D., Greenberg, L., Mashkif, N., Rinat, R.: Pluggable analysis viewpoints for design space exploration. Procedia Computer Science **16**, 226 – 235 (2013)
63. Maydanchik, A.: Data Quality Assessment. Technics Publications, LLC, Bradley Beach, NJ, USA (2007)
64. Meloni, P., Capotondi, A., Deriu, G., Brian, M., Conti, F., Rossi, D., Raffo, L., Benini, L.: Neuraghe: Exploiting cpu-fpga synergies for efficient and flexible cnn inference acceleration on zynq socs. CoRR **abs/1712.00994** (2017)
65. Meloni et al: ALOHA: an architectural-aware framework for deep learning at the edge. In: Proceedings of the Workshop on INTelligent Embedded Systems Architectures and Applications - INTESA. pp. 19–26. ACM Press (2018)
66. Meloni et al: Architecture-aware design and implementation of CNN algorithms for embedded inference: the ALOHA project. In: 2018 30th International Conference on Microelectronics (ICM). pp. 52–55 (2018)
67. Meloni et al: Optimization and deployment of cnns at the edge: The ALOHA experience. In: Proceedings of the 16th ACM International Conference on Computing Frontiers. p. 326–332. CF '19 (2019)
68. Menzies, T., Milton, Z., Turhan, B., Cukic, B., Jiang, Y., Bener, A.: Defect prediction from static code features: current results, limitations, new approaches. Automated Software Engineering **17**(4), 375–407 (2010)

69. Moser, M., Pichler, J., Fleck, G., Witlatschil, M.: Rbg: A documentation generator for scientific and engineering software. In: 2015 IEEE 22nd International Conference on Software Analysis, Evolution, and Reengineering (SANER). pp. 464–468. IEEE (2015)
70. Méhes, G., Luegmayr, A., Kornmüller, R., Ambros, I.M., Ladenstein, R., Gadner, H., Ambros, P.F.: Detection of disseminated tumor cells in neuroblastoma: 3 log improvement in sensitivity by automatic immunofluorescence plus FISH (AIPF) analysis compared with classical bone marrow cytology. The American Journal of Pathology **163**(2), 393–399 (2003)
71. Newman, S.: Building Microservices. O'Reilly Media, Inc., 1st edn. (2015)
72. Nickel, M., Murphy, K., Tresp, V., Gabrilovich, E.: A review of relational machine learning for knowledge graphs. Proceedings of the IEEE **104**(1), 11–33 (2016)
73. Nielson, F., Nielson, H.R., Hankin, C.: Principles of program analysis. Springer (2015)
74. Nikzad-Langerodi, R., Zellinger, W., Lughofer, E., Saminger-Platz, S.: Domain-invariant partial-least-squares regression. Analytical chemistry **90**(11), 6693–6701 (2018)
75. Noy, N., Gao, Y., Jain, A., Narayanan, A., Patterson, A., Taylor, J.: Industry-scale knowledge graphs: Lessons and challenges. Communications of the ACM **62 (8)**, 36–43 (2019)
76. Obermeyer, Z., Powers, B., Vogeli, C., Mullainathan, S.: Dissecting racial bias in an algorithm used to manage the health of populations. Science **366**(6464), 447–453 (2019)
77. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Transactions on knowledge and data engineering **22**(10), 1345–1359 (2009)
78. Pascarella, L., Bacchelli, A.: Classifying code comments in java open-source software systems. In: 2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR). pp. 227–237. IEEE (2017)
79. Paulheim, H.: Knowledge graph refinement: A survey of approaches and evaluation methods. Semantic web **8**(3), 489–508 (2017)
80. Pimentel, A.D., Erbas, C., Polstra, S.: A systematic approach to exploring embedded system architectures at multiple abstraction levels. IEEE Trans. Comput. **55**(2), 99–112 (2006)
81. Quionero-Candela, J., Sugiyama, M., Schwaighofer, A., Lawrence, N.D.: Dataset shift in machine learning. The MIT Press (2009)
82. Ramler, R., Buchgeher, G., Klammer, C.: Adapting automated test generation to gui testing of industry applications. Information and Software Technology **93**, 248–263 (2018)
83. Ramler, R., Felderer, M.: A process for risk-based test strategy development and its industrial evaluation. In: International Conference on Product-Focused Software Process Improvement. pp. 355–371. Springer (2015)
84. Ramler, R., Wolfmaier, K.: Issues and effort in integrating data from heterogeneous software repositories and corporate databases. In: Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement. pp. 330–332 (2008)
85. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. pp. 234–241 (2015)
86. Samek, W., Wiegand, T., Müller, K.R.: Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. arXiv e-prints (2017)

87. Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.F., Dennison, D.: Hidden technical debt in machine learning systems. In: 28th International Conference on Neural Information Processing Systems (NIPS). pp. 2503–2511 (2015)
88. Sebastian-Coleman, L.: Measuring Data Quality for Ongoing Improvement. Elsevier (2013)
89. Shinyama, Y., Arahori, Y., Gondow, K.: Analyzing code comments to boost program comprehension. In: 2018 25th Asia-Pacific Software Engineering Conference (APSEC). pp. 325–334. IEEE (2018)
90. Skala, K. (ed.): Explainable Artificial Intelligence: A Survey. Croatian Society for Information and Communication Technology, Electronics and Microelectronics - MIPRO (2018)
91. Sobieczky, F.: An interlacing technique for spectra of random walks and its application to finite percolation clusters. Journal of Theoretical Probability **23**, 639—670 (2010)
92. Sobieczky, F.: Bounds for the annealed return probability on large finite percolation graphs. Electron. J. Probab. **17**, 17 pp. (2012)
93. Sobieczky, F.: Explainability of models with an interpretable base model: explainability vs. accuracy (2019), symposium on Predictive Analytics 2019, Vienna
94. Steidl, D., Hummel, B., Juergens, E.: Quality analysis of source code comments. In: 2013 21st International Conference on Program Comprehension (ICPC). pp. 83–92. Ieee (2013)
95. Sun, B., Saenko, K.: Deep coral: Correlation alignment for deep domain adaptation. In: Workshop of the European Conference on Machine Learning. pp. 443–450. Springer (2016)
96. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks. arXiv e-prints (2013)
97. Sünderhauf, N., Brock, O., Scheirer, W., Hadsell, R., Fox, D., Leitner, J., Upcroft, B., Abbeel, P., Burgard, W., Milford, M., Corke, P.: The limits and potentials of deep learning for robotics. The International Journal of Robotics Research **37**(4-5), 405–420 (2018)
98. Van Geet, J., Ebraert, P., Demeyer, S.: Redocumentation of a legacy banking system: an experience report. In: Proceedings of the Joint ERCIM Workshop on Software Evolution (EVOL) and International Workshop on Principles of Software Evolution (IWPSE). pp. 33–41 (2010)
99. Vapnik, V.N.: Statistical Learning Theory. Wiley-Interscience (1998)
100. Vidal, R., Bruna, J., Giryes, R., Soatto, S.: Mathematics of deep learning. arXiv e-prints (2017), cite arxiv:1712.04741
101. Wang, Q., Mao, Z., Wang, B., Guo, L.: Knowledge graph embedding: A survey of approaches and applications. IEEE Transactions on Knowledge and Data Engineering **29**(12), 2724–2743 (2017)
102. Wang, R.Y., Strong, D.M.: Beyond Accuracy: What Data Quality Means to Data Consumers. J. of Management Information Systems (1996)
103. Wang, Y.E., Wei, G.Y., Brooks, D.: Benchmarking TPU, GPU, and CPU Platforms for Deep Learning. arXiv e-prints (2019)
104. Xu, G., Huang, J.Z.: Asymptotic optimality and efficient computation of the leave-subject-out cross-validation. Annals of Statistics **40**(6), 3003–3030 (2012)
105. Yu, T., Zhu, H.: Hyper-parameter optimization: A review of algorithms and applications. arXiv e-prints (2020)

106. Zellinger, W., Grubinger, T., Lughofer, E., Natschläger, T., Saminger-Platz, S.: Central moment discrepancy (cmd) for domain-invariant representation learning. International Conference on Learning Representations (2017)
107. Zellinger, W., Grubinger, T., Zwick, M., Lughofer, E., Schöner, H., Natschläger, T., Saminger-Platz, S.: Multi-source transfer learning of time series in cyclical manufacturing. Journal of Intelligent Manufacturing **31**(3), 777–787 (2020)
108. Zellinger, W., Moser, B.A., Grubinger, T., Lughofer, E., Natschläger, T., Saminger-Platz, S.: Robust unsupervised domain adaptation for neural networks via moment alignment. Information Sciences **483**, 174–191 (2019)
109. Zellinger, W., Moser, B.A., Saminger-Platz, S.: Learning bounds for moment-based domain adaptation. arXiv preprint arXiv:2002.08260 (2020)
110. Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O.: Understanding deep learning requires rethinking generalization. International Conference on Learning Representations (201z)
111. Zou, J., Schiebinger, L.: AI can be sexist and racist — it's time to make it fair. Nature **559**, 324–326 (2018)