Foundations of Trustworthy AI – Integrating Reasoning, Learning and Optimization

TAILOR

Grant Agreement Number 952215

Automated AI v.2

Document type (nature)	Review report
Deliverable No	**7.2**
Work package number(s)	7
Date	August 31, 2024
Responsible Beneficiary	ULEI, ID #7
Deliverable Author(s)	Annelot Bosman, Holger Hoos, Joaquin Vanschoren
Deliverable Publicity level	Public
Short description	Version 2 of report on novel insights, techniques, algorithms and tools developed within tasks T7.1, T7.2, T7.3, T7.4 and T7.5, and how they contribute towards trustworthy AI.

Reviewers	**Institution**	**Date of approval**
Fredrik Heintz	Linkopig University	11-08-2024
Jan van Rijn	Leiden University	20-08-2024

Table of Contents

Summary of the report

This report gives an overview of the accomplishments of the AutoAI work package during the TAILOR project. The report is divided into sections according to the tasks that are part of this work package.

1. Introduction

As the practical applications of AI become more widespread, the need for skilled professionals to design, calibrate and implement these techniques continues to grow. In the field of AutoAI, research focuses on developing frameworks that automate the design, calibration and training of AI models and methods. These frameworks are intended to be reusable for various tasks, simplifying design processes and making AI more accessible to scientists and specialised workers. In recent years, they have become particularly prominent in the area of machine learning, where they are known under the label of AutoML, but also hold significance for other areas of AI, such as automated reasoning, planning, scheduling and optimisation.

Achieving the goal of a versatile, reusable framework presents several challenges. The first is determining which steps of the process should be automated in the framework. Another is how to ensure models and methods are robust and reliable and whether one wants to have an internal alarm system for when AI systems or their components are "out of their depth". Another significant challenge is minimising overhead and compute costs. At the same time, AutoAI can help achieve this by reusing information from previous tasks. Any AutoAI system should be easily adaptable and extendable.

During the TAILOR project, this work package has aimed to work on these challenges and make significant improvements. In this document, we provide a high-level overview of our achievements, insights, algorithms and tools.

This report is structured as follows; Section 2 discusses the progress in Task 2, AutoML in the wild; Section 3 looks at the progress in Task 3, Beyond standard supervised learning; Section 4 discusses Task 4, Self-monitoring AI-systems; Section 5 highlights Task 5, Multi-objective Auto-AI and finally Section 6 discusses Task 6 Ever-learning AutoAI.

2. AutoML in the wild [T7.2, ALU-FR]

Making AutoML practical in the wild necessitates pushing existing research methods and ideas to become more practical, often through practical speedups or off-the-shelf tools. To this end, we highlight some important advancements made towards bringing AutoML into the wild.

2.1 Hyperparameter Optimisation

A central part of many automated machine learning systems is to search for optimal settings for a machine learning pipeline for a given user's scenario. One new novel technique **[MalikEtAl23]** simplifies a non-experts specification of a *Bayesian Prior*, effectively incorporating a user's prior knowledge of the problem into the search procedure. However, even with this information, techniques incorporating Bayesian Optimization can often be misconfigured, leading to underwhelming utility when applied in practice. To rectify this, there are new self-adaptive techniques **[HyarfnerEtAl23]** to even further reduce the expertise required to utilise these systems.

However, in the current age of large-scale Deep Learning (DL) and Generative AI, the cost of utilising AutoML becomes prohibitive and requires novel solutions to solve these new challenges. One such interesting work **[AdriansenEtAl23]** utilises synthetic data as a prior, essentially to train a single model, *only once offline,* that can cheaply predict the learning curve of other deep-learning models. While similar techniques exist to solve this problem in an *online* fashion, this method was found to be 10,000 times faster once trained, while remaining re-usable across scenarios. From this, we developed a novel Hyperparameter Optimization technique **[RakotoarisonEtAl24]**, that pushes us closer towards AutoML for large-scale DL models.

2.2 Model Selection

While many large-scale DL models exist in the wild, very few have the capacity to train these models from scratch, nor the expertise to do so. The primary method through which these models are used is to select one from an ever-expanding pool and *fine-tune* them to a specific application or need. Given the diverse variety of such models, it is not clear which to select, let alone how to do so cost-effectively. By leveraging existing Algorithm Selection techniques and a pool of pre-trained models to choose from, we developed a practical method to greatly reduce cost and automate this process, lowering the expertise required to effectively select the appropriate model **[ArangoEtAl24]**. While this work was previously applied in a research setting, we are pushing toward making an off-the-shelf tool available to practitioners and industry.

We performed a comprehensive meta-learning study of data sets and methods for multilabel classification (MLC). MLC is a practically relevant machine learning task where each example is labelled with multiple labels simultaneously. Here, we analyse 40 MLC data sets by using 50 meta features describing different properties of the data. The main findings of this study are as follows. First, the most prominent meta features that describe the space of MLC data sets are the ones assessing different aspects of the label space. Second, the meta models show that the most important meta features describe the label space, and, the meta features describing the relationships among the labels tend to occur a bit more often than the meta features describing the distributions between and within the individual labels. Third, the optimization of the hyperparameters can improve the predictive performance, however, quite often the extent of the improvements does not always justify the resource utilisation **[BogatinovskiEtAl22].**

Furthermore, we developed a method for explainable and model-specific algorithm selection for multi-label classification. Namely, a plethora of MLC algorithms have been proposed in the literature, resulting in a meta-optimization problem that the user needs to address: which MLC approach to select for a given dataset? To address this algorithm selection problem, we investigate in this work the quality of an automated approach that uses characteristics of the datasets - so-called features - and a trained algorithm selector to choose which algorithm to apply for a given task. For our empirical evaluation, we use a portfolio of 38 datasets. We consider eight MLC algorithms, whose quality we evaluate using six different performance metrics. We show that our automated algorithm selector outperforms any of the single MLC algorithms, and this is for all evaluated performance measures. Our selection approach is explainable, a characteristic that we exploit to investigate which meta-features have the largest influence on the decisions made by the algorithm selector. Finally, we also quantify the importance of the most significant meta-features for various domains **[KostovskaEtAl22a]**.

2.3 Vertical Matchmaking

The use of Artificial Intelligence (AI) in industry has grown significantly. Techniques like Deep Learning and optimization demand substantial computational and storage resources. Choosing the right hardware (on-premise or cloud) and determining its capacity for AI algorithms is crucial yet challenging, especially when quality-of-service constraints or budgets are involved. The possibility of deploying AI models on a variety of hardware devices with different computing resources according to specific computing tasks (e.g., performing inference with DNNs) makes the problem even more complicated, even for experts. An automated decision support tool to match algorithms, user constraints, and hardware resources would greatly benefit companies and practitioners. We proposed a data-driven approach to assist AI adopters and developers in choosing the optimal hardware resource and hyperparameters configuration of a given AI algorithm.

Our approach is based on three key elements: i) fair benchmarking of target AI algorithms on a set of heterogeneous platforms, ii) the creation of ML models to learn the behaviour of these AI algorithms, and iii) support guidelines to help identify the best deployment option for a given AI algorithm.

We employ the Empirical Model Learning paradigm, which integrates Machine Learning (ML) models into an optimisation problem. This approach combines expert domain knowledge with data-driven models to learn the relationships between hardware requirements and AI algorithm performance. We benchmark multiple AI algorithms on various hardware resources to generate data for training ML models, then use optimization to find the best hardware configuration that meets user-defined constraints like budget, time, and solution quality **[FrancobaldiEtAl23, De FilippoEtAl22].**

2.4 AutoML for Satellite Imagery

We developed a Neural Architecture Search (NAS) framework for super-resolution Earth Observation (EO) satellite images. Satellites orbiting the Earth continually collect data about our atmosphere, oceans, and land. These satellite images have many high-impact applications with machine learning potential, like weather prediction, deforestation detection, and crop monitoring. However, the effort required for manual design and configuration of machine learning EO pipelines creates a bottleneck in our ability to create solutions for these highly relevant problems. Therefore we make state-of-the-art machine learning models accessible to EO domain researchers by automating the design of models for EO tasks. For example, we developed a Neural Architecture Search (NAS) framework for super-resolution for EO images. Super-resolution is a pre-processing step that increases the resolution of satellite images. This is important for tasks where detailed information is highly relevant, such as change detection. We propose a search space based on state-of-the-art super-resolution networks. We demonstrate the adaptability of our approach on four satellite image datasets, including a novel dataset we collected. Our approach, AutoSR4EO, was published in the Remote Sensing journal.

3. Beyond standard supervised learning [T7.2, ULEI]

3.1. AutoML for unsupervised learning via meta-learning

Unsupervised learning tasks, such as clustering and outlier detection, have long been eluding AutoML research, since AutoML usually needs a ground truth (a golden standard) to get a signal whether one model is better than another. This is usually not available in unsupervised problems. However, human experts do not have difficulties proposing good unsupervised learning algorithms based on prior experience. TUE developed a way to emulate this via meta-learning. By recording the performance of (many variations of) unsupervised techniques on many prior problems, we can recommend the best techniques based on how similar a new problem is to prior ones. This similarity can be accurately measured using ‘optimal transport’, a technique to measure the similarity between two data distributions. Indeed, if two data distributions are very similar, then the same unsupervised techniques will likely work well, which we also demonstrated empirically on large benchmarks of unsupervised tasks. This novel approach was published in two papers for different unsupervised tasks: for outlier detection at IJCAI 2023 **[SinghVanschoren23a]**, and for clustering at the NeurIPS 2023 workshop on optimal transport **[SingVanschoren23b]**.

3.2 Meta-learning for few shot learning

We developed empirical intuitions and insights on how various meta-learning and transfer learning techniques worked, and why certain techniques outperform others in certain situations. First, we surveyed the literature and described the existing methods in a common framework. Additionally, we have gained an understanding why (despite its lower complexity) the common method MAML outperforms the more expressive method Metalearner-LSTM, and have presented a method that combines the best of both (at the cost of additional

runtime). Afterwards, we did an empirical study that compares various seminal methods (i.e. MAML, Reptile and Transfer Learning) with each other across scenarios, and determines when each of these methods is beneficial to use. Finally, we showed why vanilla LSTMs are not good at few shot learning, and propose an architecture that can do this **[HuismanEtAl21, HuismanEtAl22, HuismanEtAl23, HuismanEtAl24]**.

3.3 MetaDL competition series

Finally, we ran a series of competitions, which we refer to as the MetaDL-competition series, which was aimed to further advance the state-of-the-art in meta-learning research. The series consisted of two competitions, i.e. a preliminary competition that ran in the fall of 2020 that was presented at AAAI 2021 **[ElBazEtAl22a]**, and a version that ran across 2021, and was presented at NeurIPS 2021 **[ElBazEtAl22b]**. These competitions formed the basis for the Meta-Album benchmark suite, presented further. Competition reports:

4. Self-monitoring AI systems [T7.3, ULEI]

4.1. A comprehensive survey paper of AutoML

Automated machine learning (AutoML) is a young research area aiming at making high-performance machine-learning techniques accessible to a broad set of users. This is achieved by identifying all design choices in creating a machine-learning model and addressing them automatically to generate performance-optimised models. In our recent publication, we have provided an extensive overview of the past and present, as well as future perspectives of AutoML **[BaratchiEtAl24]**.

4.2. Algorithm-Agnostic Uncertainty Estimation

Detecting and signalling when a machine learning algorithm, such as a classifier, makes erroneous predictions is of crucial importance for the development of trustworthy AI systems. At the same time, most work on uncertainty estimation tends to be limited to a specific type of predictor or only considers regression tasks. In this work, we present a flexible method for estimating uncertainty in classification procedures using several, classifier-independent measures that act as proxies for the uncertainty associated with predictions. Our approach yields promising results on a variety of benchmark datasets and can be used alone or in combination with the uncertainty estimates produced by the classifier **[KönigEtAl20]**.

5. Multi-objective AutoAI [T7.4, INRIA]

5.1. A Multi-Objective AutoML Approach for Federated Learning

Federated learning is a training paradigm according to which a server-based model is cooperatively trained using local models running on edge devices and ensuring data privacy. Keeping private data in the edge devices is an important approach for people to trust in federating learning. Edge and server devices exchange information that induces a substantial communication load, which jeopardises the functioning efficiency. The difficulty of reducing this overhead stands in achieving this without decreasing the model's efficiency. Many works investigated the compression of the pre/mid/post-trained models and the communication rounds separately, although they jointly contribute to communication overload. Our work aims at optimising the hyper-parameters of the federating learning scheme by using a multi-objective formulation where both, the accuracy and the communication overhead are simultaneously optimised.

In this paper **[MorelIEtAl22]**, we worked in a first approach using NSGA-II to tune the quantisation, and sparsification of the models, as well as the number of communicating rounds and the number of clients. A more advanced approach taking into account the heterogeneity of the edge devices and tuning more parameters was published **[MorelIEtAl24]**. In this second work, the batch size, the learning rate, the aggregation method, the decision of the information to send to the server, and the strategy to deal with zeroes are also tuned during the optimization.

5.2. A Differential Method for Multi-Objective Neural Architecture Search.

One evolving use case for Neural Architecture Search (NAS) to find well-performing models that can also be deployed on edge devices, often subject to energy or hardware constraints. However, these techniques are often cost-prohibitive, especially when considering an ever-increasing number of objectives. To accelerate this procedure, we utilise the fact that neural architectures consist of differentiable operations, to effectively explore the Pareto-frontier of candidate models that are possible given a given search space **[SukthankerEtAl24]**. This was showcased across 19 different hardware devices, both GPU and CPU, as well as for 3 objective showcases, showing to outperform existing multi-objective methods in this regime.

5.3 Enhancing Algorithm Selection and Performance Prediction in Black-Box Optimization Using Landscape Features and Machine Learning Techniques

We performed three studies to delve into enhancing the performance prediction and selection of modular optimisation algorithms using various methodologies. The first study explores the significance of landscape features in predicting the performance of modular CMA-ES variants, revealing that although the most relevant features remain consistent across different module configurations, their influence on regression accuracy varies. The second study compares machine learning models for algorithm selection (AS) in black-box optimization, demonstrating that while AS has impressive potential, the choice of the ML

model (Random Forest, XGBoost, Transformers, etc.) has minimal impact on performance. The third study introduces a methodology to distinguish easily solvable problem instances from challenging ones based on an algorithm's performance footprint, linking these instances to specific landscape properties using meta-representations and model explainability techniques **[NikolikjEtAl23, KostovskaEtAl23a, KostovskaEtAl22b]**.

5.4 OPTImization algorithm benchmarking ONtology

We focused on the development and application of ontologies for improving the benchmarking and performance prediction of optimization algorithms. The first paper introduces OPTION (OPTImization algorithm benchmarking ONtology), a semantically rich, machine-readable data model designed to enhance the interoperability, automatic data integration, and querying capabilities of different benchmarking platforms. By annotating and querying benchmark performance data from the BBOB and YABBOB collections, and integrating these into the IOHprofiler environment, OPTION facilitates meta-analysis of performance data. The second paper builds on the OPTION ontology, extending it to represent modular black-box optimization algorithms and creating knowledge graphs with performance data for modCMA and modDE frameworks. Using a knowledge graph embedding-based methodology, we demonstrate that triple classification can accurately predict whether an algorithm instance will achieve a specific target precision, highlighting the potential of this approach and calling for community collaboration on algorithm feature representation [**KostovskaEtAl23b, KostovskaEtAl23c**].

5.5 Hyperparameter Optimization for Robustness Verification

Despite their great success in recent years, neural networks are vulnerable to adversarial attacks. These attacks are often based on slight perturbations of given inputs that cause them to be misclassified. Several methods have been proposed to formally prove the robustness of a given network against such attacks. However, these methods typically give rise to high computational demands, which severely limit their scalability. Recent state-of-the-art approaches state the verification task as a minimisation problem. These are highly complex methods, solving instances of problems that have been proven to be NP-hard. In this line of research, we developed automated hyperparameter optimization methods, so that these methods are more efficient and can verify larger network types **[KönigEtAl21,KönigEtAl22,KönigEtAl23,KönigEtAl24a,KönigEtAl24b]** .

Other output modalities:

- Podcast episode: Jan van Rijn: Robustness, unveiling the black box of AI (Computers Don't Byte)

5.6 Neural architecture search focussing on accuracy and robustness

Neural networks are vulnerable to slight alterations to correctly classified inputs, leading to incorrect predictions. To rigorously assess the robustness of neural networks against such

perturbations, formal verification techniques can be employed. Robustness is generally measured in terms of adversarial accuracy, based on an upper bound on the magnitude of perturbations commonly denoted as epsilon. This complicates the neural architecture search to a multi-objective optimisation problem, where we not only want to optimise for accuracy, but also for robust accuracy. Due to the complexity of this problem, we have not developed methods that solve this multi-objective problem yet. However, we have developed measures that can express these measures.

Some of these measures are:

- The critical epsilon values are a per instance measure for the degree towards which an instance is robust. Critical epsilon values can be used to create reliable so-called robustness distributions, that give an empirical distribution of the robustness of a neural network across instances **[BosmanEtAI23]** with an extended version under submission 2024.

- The robustness distributions can be further refined to not only take into consideration robustness across all instances of a dataset, but also robustness on a per-class level. This has important implications for label bias and in the long run also for fairness-aware systems **[BosmanEtAI24]**.
- The delta-values are a per-instance proxy for the critical epsilon values. While the correlation with the critical epsilon value is low, it can be used to empower racing methods to select a network from a set of options efficiently **[KönigEtAI24c]**.

5.7 Streaming machine translation

Streaming Machine Translation (MT) is the task of translating an unbounded input text stream in real-time. The traditional cascade approach, which combines an Automatic Speech Recognition (ASR) and an MT system, relies on an intermediate segmentation step which splits the transcription stream into sentence-like units. However, the incorporation of a hard segmentation constrains the MT system and is a source of errors. We have proposed a Segmentation-Free framework that enables the model to translate an unsegmented source stream by delaying the segmentation decision until the translation has been generated. Extensive experiments show how the proposed Segmentation-Free framework has better quality-latency trade-off than competing approaches that use an independent segmentation model **[Iranzo-SánchezEtAI24]**.

6. Ever-learning AutoAI [T7.5, TU/e]

Many problems in trustworthy AI can only really be addressed by collaborating on a global scale and by delivering high-quality tools that everyone can use. To gain adoption in benchmarking, it is important to work together with the authors of all the leading algorithms to make sure that they are used correctly and to test all methods on a very large set of public datasets assembled together with the community. If the results are hard to dispute and

accepted by the major stakeholders, a benchmark can quickly gain traction. Moreover, for meta-learning, experiments need to be run across even larger sets of models and datasets, which requires very careful organisation of all resources and results, something that no single lab can do by itself. By building and working with the OpenML community, and collaborating with other initiatives with shared goals, we were able to have much more impact than would otherwise be possible. This is underscored by the following outcomes:

6.1 Benchmarks

OpenML Benchmarking Suites: A method to easily create new curated benchmarks across many datasets under very specific constraints, so that the results are easily interpretable and reproducible. Published at NeurIPS, and leading to many new benchmark suites being proposed (27 of which are also on OpenML) **[BischlEtAl21,UllahEtAl22]** .

Moreover, these benchmarking suites have already led to more than 120 new papers, many of which have very interesting and novel findings. For instance, these benchmarks showed that tree-based models still outperform deep learning models on tabular data **[GrinsztajnEtAl22]**, which stimulated research into novel deep learning techniques, especially transformer-based methods, for tabular data. Well-known examples of this are **[HollmannEtAl22]**, which use in-context learning on tabular data, and foundational hypernetworks **[MüllerEtAl23]**, both of which reached state-of-the-art results on tabular data.

The AutoML Benchmark: an extensible architecture to systematically benchmark AutoML frameworks (built upon OpenML Benchmarking Suites). Adopted by 15 leading AutoML frameworks, many of which are from industry, and contributed by the original authors. This includes AutoGluon (Amazon), AutoSKLearn (U Freiburg), GAMA (U Eindhoven), H2O-AutoML (H2O), ML.NET AutoML (Microsoft), Auto-XGBoost and MLR3AutoML (U Munich), FLAML (Microsoft), MLJAR-AutoML (MLJAR), OBOE (Cornell), LightAutoML (Sberbank AI), hyperopt-sklean (U Waterloo), and MLPlan (U Paderborn). Published in JMLR. Together with the ArXiv version, cited more than 320 times **[GijsbersEtAl24]**.

Croissant: a meta-data format for machine learning datasets, making them more portable and better described to allow easier exchange and usage. It is supported by OpenML, HuggingFace, Kaggle, Google Dataset Search, and TensorFlow Datasets. Is it also being adopted by Harvard Dataverse, and the NeurIPS conference is recommending it to be used for all new datasets submitted to the conference. This work also won the best paper award at the DEEM Workshop at SIGKDD 2024 **[AkhtarEtAl24]**.

Finally, by reasoning about how human machine-learning experts use their experience across many prior tasks, we learned how to apply meta-learning to solve problems in new and interesting ways. To highlight a few:

- We can **more accurately recommend** unsupervised learning techniques (e.g. for clustering and outlier detection) by estimating how similar new tasks are to older tasks for which we know effective solutions **[SinghVanschoren23]**

- We can **more efficiently tune** machine learning algorithms by pre-training a deep learning meta-model that predicts which hyperparameters to try next.
- We can **fine-tune models continually** by replacing gradient descent with a meta-learning transformer-based optimizer that learns which weights in the network should be updated to learn fast, and which to leave untouched to avoid forgetting previous tasks.

6.2 Towards AutoAI for Neurosymbolic AI

A prominent field that combines learning and reasoning is the field of neuro-symbolic AI (NeSy), cf. WP 4, which combines logic, probability theory, and deep neural networks. Many NeSy systems have properties that make them suitable for AutoAI: they are declarative in nature, can combine multiple ML learning models, and generalise well between tasks. However, the current state of the field is almost antithetical to the goals of AutoAI. State-of-the-art systems are complex, have diverse interfaces, and often require considerable effort from experts to deploy them onto new tasks. In fact, it is already non-trivial to compare a wide variety of systems on a consistent set of benchmarks.

Researchers at the KU Leuven published two works to tackle this discrepancy **[VermeulenEtAL23]**. In this work, the authors provide an overview and categorization of popular state-of-the-art NeSy systems and a classification of the benchmarks they are applied on. From these results, the authors conclude that the field of neuro-symbolic AI is segmented and that each segment has its own set of benchmark tasks that is used for evaluation, hindering a fair and complete comparison among systems. In this work, they extend the experimental comparison towards what was missing in the literature, showcasing the strengths and weaknesses of different types of systems.

One of the main issues in NeSy is that every system has its own unique syntax to represent the knowledge used in the model **[KriekenEtAL24]**. In this work, the authors take a first step towards providing a uniform representational language for neuro-symbolic knowledge, called ULLER. It is an extension of first-order logic that includes special considerations NeSy settings. The exact semantics of the language are also specified separately as the semantics of these NeSy systems are not uniform.

One of the arguably most interesting classes of neurosymbolic architectures are systems that perform *program synthesis*. Such architectures, given an input problem (e.g. represented with a set of examples), synthesise a program in a bespoke Domain-Specific Language (DSL) that expresses a solution to that given problem. The program obtained in this way can be then executed, i.e. applied to the input problem in an attempt to solve it. While program synthesis has been the subject of intense research for a few decades, it is only in recent years that the progress in neurosymbolic systems made it more practically feasible.

The PUT team published the following works that engage program synthesis **[BednarekKrawiec24]**. In this paper, we propose a modular neural symbolic architecture for solving abstract problems based on neural program synthesis and conduct a comprehensive

analysis of decisions made by the generative module of the proposed architecture. At the core of the method is a typed DSL designed to facilitate feature engineering and abstract reasoning. In training, we use the programs that failed to solve tasks to generate new tasks and gather them in a synthetic dataset. As each synthetic task created in this way has a known associated program (solution), the model is trained on them in supervised mode. Solutions are represented in a transparent programmatic form, which can be inspected and verified. We demonstrate the performance of the method using the well-known Abstract Reasoning Corpus benchmark by F.Chollet, for which our framework generates tens of thousands of synthetic problems with corresponding solutions and facilitates systematic progress in learning.

[KrawiecEtAl24] In this work, we proposed a neural symbolic architecture that uses a DSL to capture selected priors of image formation, including object shape, appearance, categorisation, and geometric transforms. We express template programs in that language and learn their parameterisation with features extracted from the scene by a convolutional neural network. When executed, the parameterized program produces geometric primitives which are rendered and assessed for correspondence with the scene content and trained via auto-association with gradient. We confront our approach with a baseline method on a synthetic benchmark and demonstrate its capacity to disentangle selected aspects of the image formation process, learn from small data, correct inference in the presence of noise, and out-of-sample generalisation.

The experience acquired in this research activity can be transferred to AutoAI and AutoML tasks. We are particularly interested in continuing this research direction by developing AutoAI/ML methods that express ML/AI architectures as programs in a bespoke DSL, where the instructions of the DSL implement composable processing steps used in AI/ML architectures, like data preprocessing and cleaning, transformations of representations, inference models and post-processing.

7. Conclusions

During the TAILOR project, this work package has made significant progress in addressing gaps within the AutoAI community. The impact of this effort is mirrored in the numerous high-tier publications, such as "AMLB: an AutoML Benchmark" published at JMLR, and prestigious best paper awards, such as the AAAI SafeAI Workshop best paper award for “Critically Assessing the State of the Art in CPU-based Local Robustness Verification” .

Major challenges still remain. Technical aspects, such as ensuring ease of use and accessibility of tools that embody all the basic research results achieved in this project, are hard to develop. Convincing practitioners to use AutoAI tools, rather than creating complex AI tools based on domain knowledge, has proven to be highly non-trivial. Working even more closely together with practitioners and understanding domain-specific tasks will be important for achieving higher TRLs than those targeted within TAILOR. Another issue is the assurance of safety, robustness and fairness. A rudimental issue is the lack of generalisable objectives for which AI models can be trained and tested. Much effort is currently being invested by the community into progress in this direction, but as these topics are so complex, more time, effort and resources will be needed to find generally accepted and broadly usable solutions.

Overall, while substantial progress has been made, ongoing collaboration and innovation are essential to fully realise the potential of AutoAI in Europe. As a result of the work within TAILOR, the already strong European community in this area is clearly positioned for further success, which we are deeply convinced will greatly contribute to the success of "AI made in Europe".

8. References

[AdriansenEtAl23] Adriaensen, S., Rakotoarison, H., Müller, S., & Hutter, F. (2024). Efficient bayesian learning curve extrapolation using prior-data fitted networks. *Advances in Neural Information Processing Systems*, *36*.
[AkhtarEtAl24] Akhtar, M., Benjelloun, O., Conforti, C., Gijsbers, P., Giner-Miguelez, J., Jain, N., ... & Wu, C. J. (2024, June). Croissant: A Metadata Format for ML-Ready Datasets. In *Proceedings of the Eighth Workshop on Data Management for End-to-End Machine Learning* (pp. 1-6).
[ArangoEtAl24] Arango, S. P., Ferreira, F., Kadra, A., Hutter, F., & Grabocka, J. (2023). Quick-tune: Quickly learning which pretrained model to finetune and how. *arXiv preprint arXiv:2306.03828*.
[BaratchiEtAl24] Baratchi, M., Wang, C., Limmer, S., van Rijn, J. N., Hoos, H., Bäck, T., & Olhofer, M. (2024). Automated machine learning: past, present and future. *Artificial Intelligence Review*, *57*(5), 1-88.
[BednarekKrawiec24] Jakub Bednarek, Krzysztof Krawiec, Learning to Solve Abstract Reasoning Problems with Neurosymbolic Program Synthesis and Task Generation, NeSy 2024.

[BischlEtAl21] Bischl B, Casalicchio G., Feurer M., Gijsbers P., Hutter F., Lang M., Gomes Mantovani R., van Rijn, J,N., Vanschoren. J. OpenML Benchmarking suites. Advances in Neural Processing Systems, Datasets and Benchmarks (NeurIPS 2021)
[BogatinovskiEtAl22] Bogatinovski, J., Todorovski, L., Džeroski, S., & Kocev, D. (2022). Explaining the performance of multilabel classification methods with data set properties. *International Journal of Intelligent Systems*, *37*(9), 6080-6122.
[BosmanEtAl23] Bosman, A. W., Hoos, H. H., & van Rijn, J. N. (2023). A preliminary study of critical robustness distributions in neural network verification. In *Proceedings of the 6th workshop on formal methods for ML-enabled autonomous systems*.
[BosmanEtAl24] Bosman, A. W., Münz, A. L., Hoos, H. H., & van Rijn, J. N. (2024, July). A Preliminary Study to Examining Per-class Performance Bias via Robustness Distributions. In *International Symposium on AI Verification* (pp. 116-133). Cham: Springer Nature Switzerland.
[De FilippoEtAl22] De Filippo A, Borghesi A, Boscarino A, Milano M. HADA: An automated tool for hardware dimensioning of AI applications. Knowledge-Based Systems. 2022 Sep 5;251:109199.
[ElBazEtAl22a] A El Baz, I Guyon, Z Liu, JN van Rijn, S Treguer, J Vanschoren, Advances in MetaDL: AAAI 2021 challenge and workshop. AAAI Workshop on Meta-Learning and MetaDL Challenge, 1-16, 2022
[ElBazEtAl22b] A El Baz, I Ullah, E Alcobaça, ACPLF Carvalho, H Chen, F Ferreira, H Gouk, C Guan, I Guyon, T Hospedales, S Hu, M Huisman, F Hutter, Z Liu, F Mohr, E Öztürk, JN van Rijn, H Sun, X Wang, W Zhu, Lessons learned from the NeurIPS 2021 MetaDL challenge: Backbone fine-tuning without episodic meta-learning dominates for few-shot learning image classification, NeurIPS 2021 Competitions and Demonstrations Track, 2022
[FrancobaldiEtAl23] Francobaldi M, De Filippo A, Borghesi A, Pizˇurica N, Jovančević I, Llewellynn T, de Prado M. TinderAI: Support System for Matching AI Algorithms and Embedded Devices. InThe International FLAIRS Conference Proceedings 2023 May 8 (Vol. 36).
[GijsbersEtAl24] Gijsbers, P., Bueno, M. L., Coors, S., LeDell, E., Poirier, S., Thomas, J., ... & Vanschoren, J. (2024). Amlb: an automl benchmark. *Journal of Machine Learning Research*, *25*(101), 1-65.
[GrinsztajnEtAl22] Grinsztajn, L., Oyallon, E., & Varoquaux, G. (2022). Why do tree-based models still outperform deep learning on typical tabular data?. *Advances in neural information processing systems*, *35*, 507-520.
[HyarfnerEtAl23] Hvarfner, C., Hellsten, E., Hutter, F., & Nardi, L. (2024). Self-correcting bayesian optimization through bayesian active learning. *Advances in Neural Information Processing Systems*, *36*.
[HollmannEtAl22] Hollmann, N., Müller, S., Eggensperger, K., & Hutter, F. (2022). Tabpfn: A transformer that solves small tabular classification problems in a second. *arXiv preprint arXiv:2207.01848*.
[HuismanEtAl21] Huisman, M., Van Rijn, J. N., & Plaat, A. (2021). A survey of deep meta-learning. *Artificial Intelligence Review*, *54*(6), 4483-4541.
[HuismanEtAl22] Huisman, M., Plaat, A., & van Rijn, J. N. (2022). Stateless neural meta-learning using second-order gradients. *Machine Learning*, *111*(9), 3227-3244.
[HuismanEtAl23] Huisman, M., Moerland, T. M., Plaat, A., & van Rijn, J. N. (2023). Are LSTMs good few-shot learners?. *Machine Learning*, *112*(11), 4635-4662.

[HuismanEtAl24] Huisman, M., Plaat, A., & van Rijn, J. N. (2024). Understanding transfer learning and gradient-based meta-learning techniques. *Machine Learning*, *113*(7), 4113-4132.
[KönigEtAl20] König, M., Hoos, H. H., & van Rijn, J. N. (2020, July). Towards algorithm-agnostic uncertainty estimation: Predicting classification error in an automated machine learning setting. In *7th ICML Workshop on Automated Machine Learning (AutoML)*.
[KönigEtAl21] M König, HH Hoos, JN van Rijn. (2021). Speeding up neural network verification via automated algorithm configuration, ICLR Workshop on Security and Safety in Machine Learning Systems.
[KönigEtAl22] König, M., Hoos, H. H., & Rijn, J. N. V. (2022). Speeding up neural network robustness verification via algorithm configuration and an optimised mixed integer linear programming solver portfolio. *Machine Learning*, *111*(12), 4565-4584.
[KönigEtAl23] M König, A Bosman, HH Hoos, JN van Rijn, Critically Assessing the State of the Art in CPU-based Local Robustness Verification, SafeAI workshop @ AAAI, 2023. Best paper award.
[KönigEtAl24a] M König, AW Bosman, HH Hoos, JN van Rijn, Critically assessing the state of the art in neural network verification, Journal of Machine Learning Research 25 (12), 2024.
[KönigEtAl24b] M König, X Zhang, HH Hoos, M Kwiatkowska, JN van Rijn, Automated Design of Linear Bounding Functions for Sigmoidal Nonlinearities in Neural Networks, Accepted at ECML PKDD 2024 (funded by TAILOR CEF fund)
[KönigEtAl24c] König, M., Hoos, H. H., & van Rijn, J. N. (2024, March). Accelerating Adversarially Robust Model Selection for Deep Neural Networks via Racing. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 38, No. 19, pp. 21267-21275).
[KostovskaEtAl22a] Kostovska, A., Doerr, C., Džeroski, S., Kocev, D., Panov, P., & Eftimov, T. (2022, December). Explainable model-specific algorithm selection for multi-label classification. In *2022 IEEE Symposium Series on Computational Intelligence (SSCI)* (pp. 39-46). IEEE.
[KostovskaEtAl22b] Kostovska, A., Vermetten, D., Džeroski, S., Doerr, C., Korosec, P., & Eftimov, T. (2022, July). The importance of landscape features for performance prediction of modular CMA-ES variants. In *Proceedings of the Genetic and Evolutionary Computation Conference* (pp. 648-656).
[KostovskaEtAl23a] Kostovska, A., Jankovic, A., Vermetten, D., Džeroski, S., Eftimov, T., & Doerr, C. (2023, July). Comparing algorithm selection approaches on black-box optimization problems. In *Proceedings of the Companion Conference on Genetic and Evolutionary Computation* (pp. 495-498).
[KostovskaEtAl23b] Kostovska, A., Vermetten, D., Doerr, C., Džeroski, S., Panov, P., & Eftimov, T. (2021, July). Option: optimization algorithm benchmarking ontology. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion* (pp. 239-240).
[KostovskaEtAl23c] Kostovska, A., Vermetten, D., Džeroski, S., Panov, P., Eftimov, T., & Doerr, C. (2023, April). Using knowledge graphs for performance prediction of modular optimization algorithms. In *International Conference on the Applications of Evolutionary Computation (Part of EvoStar)* (pp. 253-268). Cham: Springer Nature Switzerland.

[KrawiecEtAl24] Krzysztof Krawiec, Antoni Nowinowski, Disentangling Visual Priors: Unsupervised Learning of Scene Interpretations with Compositional Autoencoder, NeSy 2024
[KriekenEtAl24] van Krieken, E., Badreddine, S., Manhaeve, R., & Giunchiglia, E. (2024). ULLER: A Unified Language for Learning and Reasoning. *arXiv preprint arXiv:2405.00532*.
[MalikEtAl23] Mallik, N., Bergman, E., Hvarfner, C., Stoll, D., Janowski, M., Lindauer, M., ... & Hutter, F. (2024). Priorband: Practical hyperparameter optimization in the age of deep learning. *Advances in Neural Information Processing Systems*, *36*.
[MorellEtAl22] Morell, J. Á., Dahi, Z. A., Chicano, F., Luque, G., & Alba, E. (2022, April). Optimising communication overhead in federated learning using NSGA-II. In *International Conference on the Applications of Evolutionary Computation (Part of EvoStar)* (pp. 317-333). Cham: Springer International Publishing.
[MorellEtAl24] Morell, J. Á., Dahi, Z. A., Chicano, F., Luque, G., & Alba, E. (2024). A multi-objective approach for communication reduction in federated learning under devices heterogeneity constraints. *Future Generation Computer Systems*, *155*, 367-383.
[MüllerEtAl23] Müller, A., Curino, C., & Ramakrishnan, R. (2023). MotherNet: A Foundational Hypernetwork for Tabular Classification. *arXiv preprint arXiv:2312.08598*.
[NikolikjEtAl23] Sukthanker, R. S., Zela, A., Staffler, B., Dooley, S., Grabocka, J., & Hutter, F. (2024). Multi-objective Differentiable Neural Architecture Search. *arXiv preprint arXiv:2402.18213*.
[RakotoarisonEtAl24] Rakotoarison, H., Adriaensen, S., Mallik, N., Garibov, S., Bergman, E., & Hutter, F. (2024). In-Context Freeze-Thaw Bayesian Optimization for Hyperparameter Optimization. *arXiv preprint arXiv:2404.16795*.
[Iranzo-SánchezEtAl24] Iranzo-Sánchez, J., Iranzo-Sánchez, J., Giménez, A., Civera, J., & Juan, A. (2023). Segmentation-Free Streaming Machine Translation. *arXiv preprint arXiv:2309.14823*.
[SinghVanschoren23a] Singh, P., & Vanschoren, J. (2023, August). AutoML for Outlier Detection with Optimal Transport Distances. In *IJCAI* (pp. 7175-7178).
[SinghVanschoren23b] Singh, P., & Vanschoren, J. Applications of Optimal Transport Distances in Unsupervised AutoML. In *NeurIPS 2023 Workshop Optimal Transport and Machine Learning*.
[SukthankerEtAl24] Sukthanker, R. S., Zela, A., Staffler, B., Dooley, S., Grabocka, J., & Hutter, F. (2024). Multi-objective Differentiable Neural Architecture Search. *arXiv preprint arXiv:2402.18213*.
[UllahEtAl22] Ullah, I., Carrión-Ojeda, D., Escalera, S., Guyon, I., Huisman, M., Mohr, F., ... & Vu, P. A. (2022). Meta-album: Multi-domain meta-dataset for few-shot image classification. *Advances in Neural Information Processing Systems*, *35*, 3232-3247.
[VermeulenEtAL23] Vermeulen, A., Manhaeve, R., & Marra, G. (2023, November). An Experimental Overview of Neural-Symbolic Systems. In *International Conference on Inductive Logic Programming* (pp. 124-138). Cham: Springer Nature Switzerland.