[RE] A Replication Study of MASIF: Meta-learned Algorithm Selection using Implicit Fidelity Information

Sara AlHasheh   *sara.alhasheh@bahcesehir.edu.tr*
Department of Artificial Intelligence Engineering
Bahçeşehir University

Hasan Aqrabawi   *hasan.aqrabawi@bahcesehir.edu.tr*
Department of Artificial Intelligence Engineering
Bahçeşehir University

Yazan ALRousan   *yazan.alrousan@bahcesehir.edu.tr*
Department of Artificial Intelligence Engineering
Bahçeşehir University

Reviewed on OpenReview: *https://openreview.net/forum?id=5aYGXxByI6*

Abstract

MASIF (Meta-Learned Algorithm Selection using Implicit Fidelity) is a framework originally introduced in Transactions on Machine Learning Research in March 2023. The way this framework works is similar to how a person would think. Imagine if you were at an open-food buffet. In all your buffet experiences, you've grown a liking to dishes containing cheese. Being at this buffet right now, you implicitly know that you're going to choose dishes containing cheese, narrowing down your choices. MASİF aims to replicate this implicit high-fidelity decision-making by leveraging transformer based models that could interpret multi-fidelity learning curve data. Hence this enables dynamically reliable, cost-considerate and informed algorithm selection without relying on parametric assumptions. Our work consists of replicating the original results via the provided datasets, and evaluate our results with respect to the initial results in the original paper. We aim to evaluate MASIF's efficiency and robustness. This project's end goal is optimizing general applicability within versatile machine learning tasks.

1 Datasets

In order to assess each dataset, we split them in each and every respective benchmark in training, validation, and test data. Afterwards, we train the corresponding algorithms on the training data under each fidelity. Finally, we compute their validation and test performance corresponding to the model trained for that fidelity. We provide partial validation curves to SH (Single-Headed learning curves) and MASIF to train on, but measure the final regret with respect to the full fidelity test performance. The same procedure is applied when the approaches are applied to new datasets. At last, we assess the performance of each approach by performing a 10-fold outer cross-validation of the meta-dataset. We use 4 different bench marks.

1.1 Synthetic:

The synthetic dataset is based on parametric learning curves taken from Mohr et al., (2022), specifically to demonstrate the myopia of SH and learning curve predictors. With the introduction of noisy curves, we can use the crossing points provided by these curves to show how MASIF alleviats myopia through strong prior knowledge with respect to the functional family an algorithm is adhering to.

1.2 Task-set:

Task-set is a dataset based on parameterization of the Adam optimizer (Kingma BA,2015), with a variety of learning rates. It runs on a large variety of modern deep learning architectures and datasets.

1.3 Scikit-CC18:

Scikit-CC18 consists of both a dataset and an algorithm meta-feature. Hence, we can evaluate SATzilla and IMFAS on it. We have evaluated multiple scikit-learn (Pedregosa et al., 2011) algorithms on the classification benchmark OpenML-CC18 (Bischl et al., 2021b)

1.4 LCBench:

Similar to the Scikit-CC18, LCBench too consists of both a dataset and algorithm meta-feature. It is a learning curve benchmark of different funnel-shaped neural networks and hyperparameters for tabular data. We sub-sampled the set of configurations to only include 170 combinations as algorithms to be selected.

These four benchmarks differ in their availability of dataset and algorithm meta-features: while Synthetic and Task-set exhibit neither, Scikit-CC18 and LCBench exhibits both. Since Scikit-CC18 and LCBench are the only benchmarks with available dataset meta-features D, we can only evaluate SATzilla and IMFAS on them. Similarly, the dataset and algorithm meta-features can only be exploited by our architecture on these benchmarks.

2 Conference

Transactions on Machine Learning Research

3 Repository

`https://github.com/saraalhasheh/Deep-Learning-MASIF`

4 Experiments

In this section, we first introduce a new evaluation protocol for our proposed extension of the classical AS meta-data setup with added fidelity information. Then, we detail our baselines along with their perks and shortcomings, briefly describe the meta-datasets we use as benchmarks, and subsequently present our findings.

4.1 Fidelity-Slice Evaluation Protocol

To assess the ability of methods to improve performance as a function of the length of the observed learning curves, we introduce a standardized fidelity-slice evaluation protocol. This protocol demonstrates how a method's top-k regret with respect to the true ranking changes as more fidelity information becomes available. We compute the regret of $s(D)$'s prediction on the test set at gradually increasing amounts of available fidelity slices, where all algorithms are observed up to the same fidelity.

4.2 Baselines

We consider several baselines highlighting different aspects of our setup and architecture:

- **Random Baseline**: A selector that randomly guesses the ranking, providing insights into the difficulty of the ranking task and the performance scale across tasks.

- **Classical Algorithm Selection (AS)**: Demonstrates the benefit of partial learning curves as an addition to classical meta-features. We compare against SATzilla'11, a portfolio-based algorithm selector.

- **Fidelity-enabled Algorithm Selection**: Extends classical AS by considering intermediate fidelities to enhance meta-knowledge.

- **Parametric Learning Curve Predictors**: Fits parametric curves to observed learning curves and extrapolates to final performances.

- **LCNet**: Utilizes Bayesian neural networks for learning curve prediction in hyperparameter optimization tasks.

- **IMFAS (Implicit Multi-Fidelity Algorithm Selection)**: Utilizes an LSTM-based architecture to refine ranking expectations with auto-regressive fidelity slices.

- **Successive Halving (SH)**: A myopic, fidelity-aware but non-parametric method that naturally produces a ranking based on terminated algorithms.

Each baseline serves to highlight different approaches and considerations in algorithm selection with varying levels of fidelity and meta-knowledge.