SmartML: A Meta Learning-Based Framework for Automated Selection and Hyperparameter Tuning for Machine Learning Algorithms

Mohamed Maher

Supervisor: Prof. Sherif Sakr

University of Tartu, Estonia

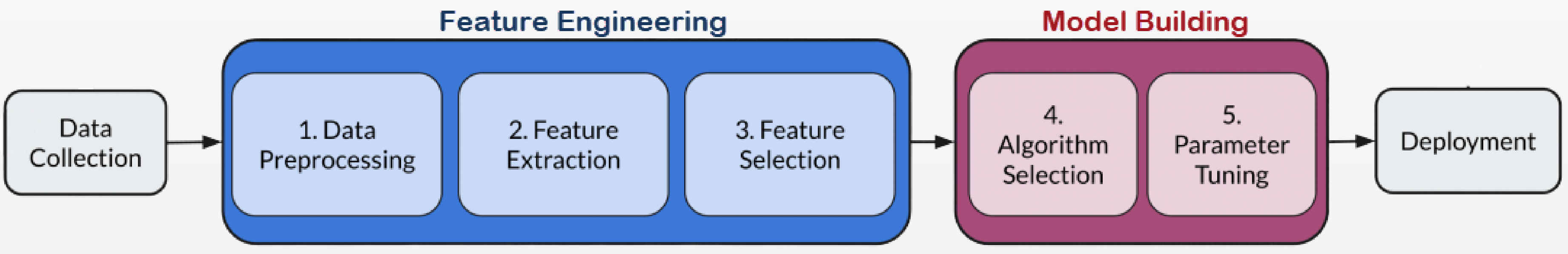
Published Paper: https://openproceedings.org/2019/conf/edbt/EDBT19_paper_235.pdf

Demo Web Interface: https://bigdata.cs.ut.ee/smartml/

Project Repository: https://github.com/mmaher22/SmartML

Introduction

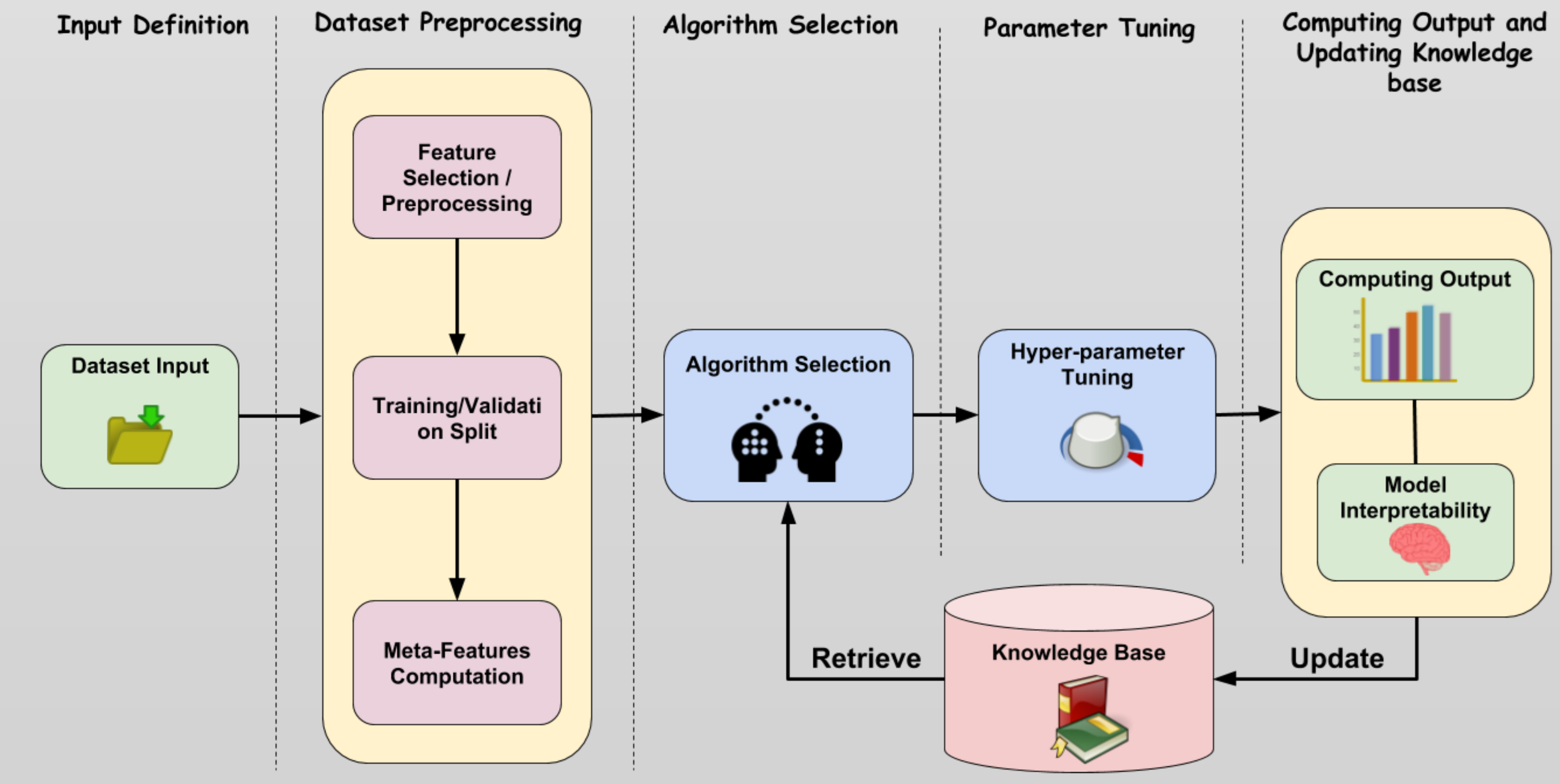
Due to the increasing success of machine learning techniques, nowadays, they have been widely utilized in almost every domain. In practice, data scientists work hard to find the best model or algorithm that meets the specifications of their problem. Such iterative and explorative nature of the modeling process is commonly tedious and time-consuming.



SmartML is a **meta learning-based framework for automated selection and hyperparameter tuning for machine learning** algorithms. Being meta learning-based, the framework is able to simulate the role of the machine learning expert. In particular, the framework is equipped with a **continuously updated knowledge base** that stores information about statistical meta-features of all processed datasets along with the associated performance of the different classifiers and their tuned parameters.

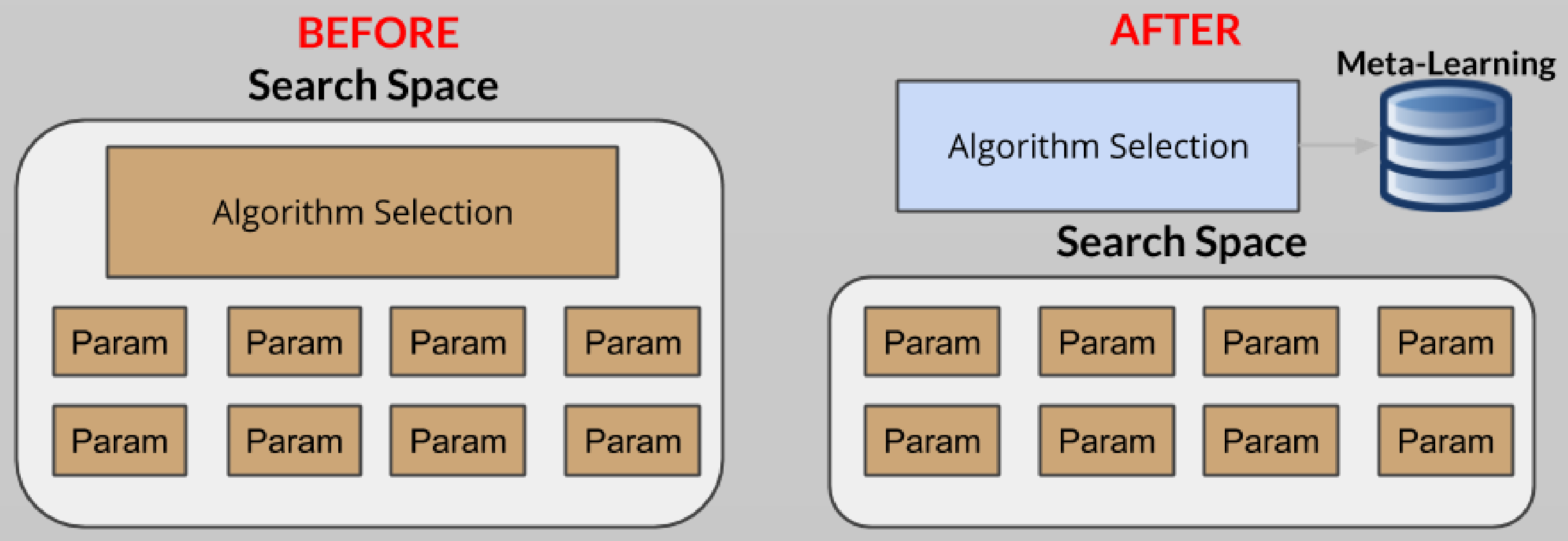
For any new dataset, **SmartML** automatically extracts its meta features and searches its knowledge base for the best performing algorithm to start its optimization process. Additionally, it makes use of the new runs to continuously enrich its knowledge base to improve its performance and robustness for future runs. The main contributions of **SmartML** are:
1. The first R-Package for automated supervised machine learning. 2. Collaborative **knowledge base (KB) for meta-learning.** 3. A modified version of Bayesian optimization with **more exploitation than exploration** for hyper-parameter tuning.

SmartML Architecture



Algorithm Selection

SmartML supports **15 different classifiers** from different R packages. In the algorithm selection phase, the meta features of the input dataset, which are extracted during the preprocessing phase, are compared with the **meta features of the datasets that are stored in the knowledge base** in order to identify the similar datasets. The dataset similarity detection process follows a weighted mechanism between two different factors. The first factors is the similarity distance between the meta-features of the dataset and meta-features of all datasets stored in the knowledge base. The second factor is the magnitude of the best performing algorithms on the similar dataset.



Hyper-Parameter Tuning

The knowledge base (KB) contains information about the best parameter configurations for each algorithm on each dataset. The configurations of the nominated best performing algorithms are used to initialize the hyper-parameter tuning process for the selected algorithms. The time budget constraint specified by the end user represents the time used in hyper parameter tuning of the selected classifiers. In particular, this budget is divided among all the selected algorithms according to the number of hyper-parameters to tune in each algorithm. SmartML applies a modified version of Sequential Model based Algorithm Configuration **(SMAC)** technique for hyperparameter optimization.

Frameworks Comparison

Framework	Auto-Weka	Auto-Sklearn	TPOT	SmartML
Language	Java	Python	Python	R
Optimization Procedure	Bayesian Optimization (SMAC / TPE)	Bayesian Optimization (SMAC)	Genetic Programming	Bayesian Optimization (SMAC)
# Classifiers Supported	27 on top of Weka	15 on top of Scikit Learn	15 on top of Scikit Learn	15 on top of R
Meta-Learning	No	Yes (Static)	No	**Yes (incrementally updated KB)**
Model Interpretability	No	No	No	**Yes**

Results

The table below shows the performance comparison between SmartML and Auto-Weka using 10 datasets where a **time budget of 10 minutes** has been allocated for each dataset in each framework. In our experiments, we have bootstrapped the knowledge base of SmartML using 50 datasets from OpenML and UCI repository.

Dataset	#Attributes	#Instances	#Classes	AutoWeka Accuracy	SmartML Accuracy
abalone	9	4177	2	25.14%	**27.13%**
amazon	10000	1500	49	57.56%	**58.89%**
cifar10small	3072	20000	10	30.25%	**37.02%**
Gisette	5000	2800	2	93.71%	**96.48%**
madelon	500	2600	2	55.64%	**73.84%**
mnist basic	784	62000	10	89.72%	**94.91%**
semeion	256	1593	10	89.32%	**94.13%**
Yeast	8	1484	10	51.80%	**66.23%**
Occupancy	5	20560	2	93.99%	**95.55%**
Kin8nm	8	8192	2	93.99%	**96.42%**

The results show that, using this relatively very small knowledge base, the accuracy results of SmartML outperform the results of Auto-Weka for all the datasets.