Universiteit Leiden
The Netherlands

Massively collaborative machine learning

Rijn, J.N. van

7 Conclusions

7.1 Open Machine Learning

We have introduced OpenML, an online platform on which researchers can share and reuse large amounts of collaboratively generated experimental data. OpenML automatically stores and indexes important meta-data about the experiments for reproducibility and further study. For datasets, data about the attributes and standard meta-features are stored. Upon these datasets, well-defined scientific tasks can be created. These provide a formal description about the given inputs and required outputs for solving it. This makes the uploaded results objectively comparable.

For the uploaded algorithms, all parameters, their data types and default values are registered. This way, it is possible to compare the performance of various algorithms, but also various parameter settings of the same algorithm. Furthermore, OpenML is integrated in popular Machine Learning workbenches and programming languages, making it possible to share algorithms and experiments with a few lines of code, or a single click of a button.

For all experiments, the exact algorithms, the inputs (such as parameter settings) and the outputs (models, predictions on test set) are stored. This makes it possible to reuse this information and learn from the past. The experimental data answers many questions about the interplay between data and algorithms, for example

- what is the best algorithm for a certain data set?
- how does a given data property influence the performance of an algorithm?
- what is the effect of a given parameter on the performance of an algorithm?
- which parameters are influencing predictive performance the most?

- which pairs of algorithms have a similar (or different) prediction behaviour?

Many of these research questions require the setup of time and computation-intensive experiments, while these can be answered on the fly when adopting this collaborative approach.

7.2 Massively Collaborative Machine Learning

We demonstrated the power of this collaborative approach by means of two large scale studies. The first study covered the data stream setting, where classifiers are continuously trained and evaluated on a stream of new observations. We ran a wide range of classifiers over all data streams in OpenML, and built meta-models to predict for each (window of) instances which classifier would work best on it. Indeed, dynamically switching at various points in the stream between various heterogeneous classifiers potentially results in a better accuracy than individual classifiers could achieve. This technique (the Meta-Learning Ensemble) indeed outperformed all individual classifiers and is competitive with state-of-the-art ensembles using many more models. Quite surprisingly, an even simpler technique that measured which of the classifiers performed best on a previous window (BLAST) outperformed all other approaches based on this idea. We introduced two variants of measuring the performance of classifiers on previous data, one approach using a fixed window and one approach based on fading factors. Furthermore, we built a clustering upon the instance-based predictions of all data stream classifiers, gaining insight in which classifiers make similar predictions (Figure 4.14 on page 69). We used OpenML to scale up data stream studies to cover 60 data streams, which is to the best of our knowledge the largest data stream study so far.

The second study covered conventional batch data, and leveraged learning curve information about algorithms. A learning curve is an ordered set of performance scores of a classifier on data samples of increasing size. OpenML contains many of these. The meta-algorithms leveraged learning curves up till a certain size, which is usually much faster than running a set of algorithms on the full dataset. Based on the performance of an algorithm on the first samples of a learning curve, assumptions can be made about the performance on the whole dataset. Within a certain budget, the most promising classifiers can be tested using a cross-validation procedure. The budget can be expressed in either an amount of cross-validation tests, or run time. In the latter case, it proved very fruitful to select classifiers based on a trade-off between accuracy and run time. If one is willing to settle for an algorithm that is almost as good as the absolute best algorithm for that dataset, the costs of finding an appropriate algorithm can be decreased by orders of magnitude. In this study, OpenML was

used as an experiment repository: the datasets and tasks that were used take many resources (time and memory) to model, so the only way to comprehensively research the proposed techniques is by reusing results that are collaboratively generated.

7.3 Community Adoption

More researchers have already adopted OpenML, as can be seen by the increasing amount of uploaded experiments. We mention some noteworthy studies, which is just a small selection of successful examples. The work of Feurer et al. [43] specializes in algorithm selection for Machine Learning by means of Sequential Model-based Bayesian Optimization. As mentioned in Chapter 3, the performance of Sequential Model-based Bayesian Optimization depends on the quality of the initial evaluation points. In order to find good initial evaluation points, the meta-knowledge in OpenML is used.

The work of Olier et al. [99] focuses on automated drug discovery. Given a protein that is critical to a pathogen (e.g. a virus or parasite), chemists are interested to know which molecules (drugs) can successfully inhibit that pathogen. This information is stored in QSAR datasets, which link the structural properties of the drugs to their activity against the protein. Machine Learning techniques can learn this relationship, but it is not clear which techniques will work best on a given QSAR dataset. The authors investigated whether a meta-algorithm can learn which techniques work well on any given QSAR dataset. All results (datasets and algorithm evaluations) are available on OpenML.

The work of Post et al. [111] uses OpenML to make general claims about common Machine Learning assumptions. In this work, the authors investigated which classifiers benefit from feature selection. Quite surprisingly, feature selection seldom led to a statistical significant improvement in performance. The authors speculate that while this might be the case, the set of datasets might be biased. As all the datasets in OpenML come from Machine Learning problems, chances are high that these already experienced some form of pre-processing. Applying these techniques on more raw data might show different results.

7.4 Future Work

With these and many other ongoing projects, there is much room for future work. Machine Learning learning literature contains a lot of 'folk knowledge' and 'folk wisdom' [38], but in order to be scientifically correct we need proper theoretical or experimental results to back these up. Much of this folklore can be confirmed or rejected

by means of large scale experimentation. Investigating questions like “is data preprocessing more important than proper algorithm selection?”, “are non-linear models really better than linear models?” and “how much additional training effort do non-linear models need to outperform their linear counterparts?” would spark interesting discussions within the community.

Another obvious possibility is to do a large scale benchmark of Machine Learning algorithms. Recently, a particular benchmark study attracted lots of attention [42], but was also criticized for various reasons [160]. In order to do proper benchmarking, the datasets, algorithms and performance space need to be properly defined. Furthermore, the relevant algorithm parameters need to be properly tuned, using for example Bayesian Optimization or Random Search. We believe that with OpenML, the infrastructure for a proper benchmark study is available, making it possible to compare algorithms across various Machine Learning toolboxes and making the results interactively available.

To conclude, there is the issue of meta-learning and optimization. As was argued in Chapter 3, there are basically two approaches to algorithm selection. The meta-learning approach learns from prior experiments, and recommends a set of classifiers based on these. The search approach experiments with intelligently trying out various classifiers (and parameter settings), but neglects the vast amount of experimental results already available. Earlier attempts to combine the two resulted in interesting ideas and solid results [43, 88], however there has been little follow-up. One possible explanation for this might be the fact that it combines two sources of knowledge that are computationally expensive to acquire, and complex to understand. OpenML partly abates both difficulties: knowledge of past experiments is available by querying the database and will lead to more understanding of both techniques. Successfully combining these two paradigms has the potential to convincingly push the state of the art of both fields of research.