

A meta-learning approach for recommending a subset of white-box classification algorithms for Moodle datasets

C. Romero, J.L. Olmo, S. Ventura
Department of Computer Science
University of Cordoba, Spain
cromero@uco.es, jlolmo@uco.es, sventura@uco.es

ABSTRACT

This paper applies meta-learning to recommend the best subset of white-box classification algorithms when using educational datasets. A case study with 32 Moodle datasets was employed that considered not only traditional statistical features, but also complexity and domain specific features. Different classification performance measures and statistics tests were used to rank algorithms. Furthermore, a nearest neighbor approach was used to recommend the subset of algorithms for a new dataset. Our experiments show that the best recommendation results are obtained when all three types of dataset features are used.

Keywords

Meta-learning, classification, predicting student performance

1. INTRODUCTION

One of the oldest and best-known problems in educational data mining (EDM) [10] is predicting student's performance as a classification task. A wide range of algorithms have been applied to predict academic success and course results. However, selecting and identifying the most adequate algorithm for a new dataset is a difficult task, due to the fact that there is no single classifier that performs best on all datasets, as proven by the No Free Lunch (NFL) theorem [6]. Choosing appropriate classification algorithms for a given dataset is of great importance in practice. Meta-learning has been used successfully to address this problem [12]. Meta-learning is the study of the main methods that exploit meta-knowledge to obtain efficient models and solutions by adapting machine learning and the DM process [4]. Recommendation can be presented in various ways, such as the best algorithm in a set, a subset of algorithms, a ranking of algorithms, or the estimated performance of algorithms. We propose to use several classification evaluation measures and statistical tests to rank algorithms, and a nearest neighbor approach to recommend the subset of best algorithms for a given new dataset.

Meta-learning has been used mainly in general domain and publicly available datasets such as UCI [2]. However, we have not found any papers that tackle algorithm selection using meta-learning in the EDM domain. There is only one related work about using meta-learning to support the selection of parameter values in a J48 classifier using several educational datasets [8]. In the educational domain, the comprehensibility of discovered classification models is an important issue, since they should be interpretable by users who are not experts in data mining (such as instructors, course authors and other stakeholders) so they can be used in decision-making processes. Indeed, white-box DM models based on rules are preferred to black-box DM models such as Bayesian and artificial neural networks, although they are

normally more accurate but less comprehensible [11]. On the other hand, statistics and information theory measures [3] and more recently data complexity measures [7] are widely used to characterize datasets in meta-learning. However, we propose to also use domain specific measures to characterize datasets.

The paper is organized as follows: Section 2 introduces the methodology used in this work; Section 3 describes the Moodle educational datasets employed in the experimentation; Section 4 describes the experiments, results, and the model obtained; and finally, conclusions and future works are outlined in Section 5.

2. METHODOLOGY

We propose a meta-learning methodology that consists of two steps (see Figure 1):

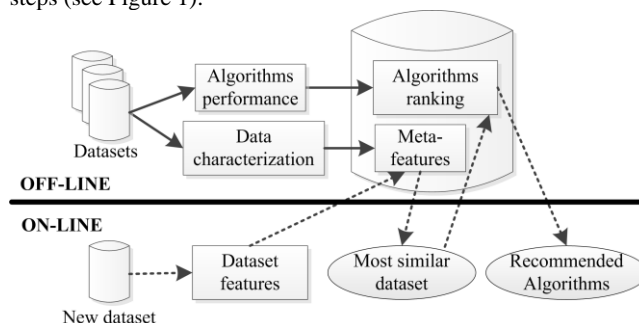


Figure 1. Meta-learning methodology.

- An off-line or training phase for creating the meta-database starting from educational datasets and classification algorithms. On the one hand, we identified important properties for characterizing datasets (statistics, complexity and domain) and developing meta-features. On the other hand, we used white-box classification algorithms (rule-based and decision tree algorithms) to evaluate their performance on all the available datasets. For each dataset, we used a statistical test on several classification evaluation measures to rank and select the subset of algorithms that gave the best performance, in such a way that there were no significant differences, as far as performance is concerned, between all the algorithms in the subset.
- An on-line or prediction phase to recommend a subset of classification algorithms to a new dataset using a nearest neighbor approach. Firstly, when a new dataset appears, its features are compared against all the meta-features in order to find the most similar dataset. Then, the subset of algorithms recommended for the new dataset corresponds to those previously obtained for its nearest neighbor.

3. DATASETS

We used a set of 32 classification educational datasets (see Table 1) about predicting students' final performance starting with Moodle's usage data [10]. As input attributes, these datasets contain a variety of information about the interaction of students in Moodle and the class to be predicted is the final mark (categorical) obtained by students in the courses. All this data was collected from university Computer Science students between 2007 and 2012. For each dataset, 16 features were obtained that can be grouped into the following three types:

- Statistical features (see Columns 2 to 6 in Table 1): the number of instances or students (Ni), the number of numerical attributes (Nna), the number of categorical attributes (Nca), the number of classes or labels of the mark attribute such as Pass/Fail, High/Medium/Low, etc. (Nc), and the imbalance ratio (IR), which is the ratio between instances of the majority class and minority class.
- Complexity features that characterize the apparent complexity of datasets for supervised learning [7], such as the maximum Fisher's discriminant ratio, the overlap of the per-class bounding boxes, the maximum (individual) feature efficiency, the collective feature efficiency (sum of each feature efficiency), the fraction of points on the class boundary, the ratio of average intra/inter class nearest neighbor distance, the leave-one-out error rate of the one-nearest neighbor classifier, the non-linearity of the one-nearest neighbor classifier, the fraction of maximum covering spheres, and the average number of points per dimension. We used DCoL (data complexity library) to obtain all the previous complexity measures [9] from our datasets.
- A domain feature (see the last column in Table 1) that indicates what the specific source of each dataset is, which can either be a Moodle's report, quiz or forum. Report is a general summary about the interaction of each student in Moodle, such as: total time in Moodle, number of accesses/sessions, number of resources viewed, number of assignments done, average score in assignments done, total time spent on assignments, number of activities carried out, total time spent on activities, etc. Quiz is a specific summary about the interaction of each student with quizzes, such as: total time spent on quizzes and each quiz done, number of quizzes answered, number of quizzes passed, average score in quizzes, correctly/incorrectly answered questions, knowledge in each concept evaluated by the quiz, etc. Forum is a specific summary about the interaction of each student with forums, such as: total time spent in forums and each forum, number of messages sent, number of messages read, number of threads created, number of replies received, number of words and sentences written, etc.

4. EXPERIMENTS

An initial experiment was carried out to select a subset of white-box classification algorithms that best predicted the final students' performance for each Moodle dataset. We used only rule-based and decision trees algorithms due to the fact that they provide models that can be easily understood by humans and used directly in the decision-making process.

Dataset	Ni	Nna	Nca	Nc	IR	Domain
Dataset1	98	4	0	2	1.08	Report
Dataset 2	194	0	4	2	1.39	Report
Dataset 3	786	6	0	3	9.8	Quiz
Dataset 4	658	0	6	3	9.1	Quiz
Dataset 5	67	40	0	2	1.23	Quiz
Dataset 6	922	6	0	3	19.27	Quiz
Dataset 7	910	0	6	3	19.24	Quiz
Dataset 8	114	0	11	2	1.19	Forum
Dataset 9	42	0	11	2	6	Forum
Dataset 10	103	0	11	2	1.53	Forum
Dataset 11	114	11	0	2	1.43	Forum
Dataset 12	98	0	6	2	1.91	Forum
Dataset 13	81	6	0	2	1.19	Forum
Dataset 14	33	0	12	2	32	Forum
Dataset 15	82	0	12	2	3.1	Forum
Dataset 16	113	40	0	4	23.5	Quiz
Dataset 17	105	41	0	3	1.06	Quiz
Dataset 18	123	0	10	4	3.89	Quiz
Dataset 19	102	10	0	3	1.06	Quiz
Dataset 20	75	0	8	2	2.12	Report
Dataset 21	52	0	4	2	1.89	Report
Dataset 22	208	10	0	2	3.25	Report
Dataset 23	438	0	10	4	15.41	Report
Dataset 24	421	10	0	4	14.2	Report
Dataset 25	84	6	0	4	5.43	Report
Dataset 26	168	6	0	4	11.25	Report
Dataset 27	136	6	0	4	11.5	Report
Dataset 28	283	0	10	2	1.67	Report
Dataset 29	155	0	10	2	1.21	Report
Dataset 30	72	6	0	4	11	Report
Dataset 31	40	0	10	2	1.2	Quiz
Dataset 32	48	10	0	2	1.8	Quiz

Table 1. Statistics and domain features of the datasets.

The next 19 classification algorithms provided by Weka 3.6 [13] were used:

- **Rule-based algorithms:** ConjunctiveRule, DecisionTable, DTNB, JRip, NNge, OneR, PART, Ridor and ZeroR.
- **Tree-based algorithms:** BFTree, DecisionStump, J48, LADTree, LMT, NBTree, RandomForest, RandomTree, REPTree and SimpleCart.

We executed each algorithm using all the Moodle datasets, which account for a total of 608 executions (19 algorithms * 32

datasets). All algorithms were executed using their default parameters and 10-fold cross-validation.

Several classification performance measures were used to compare algorithm performance [13], such as sensitivity (Sen), precision (Prec), F-Measure (F-M), Kappa (Kap) and the area under the ROC curve (AUC). For instance, Table 2 shows the average values for these measures obtained by each algorithm on dataset1.

Algorithm	Sen	Prec	F-M	Kap	AUC
RConjunctiveRule	0.845	0.869	0.846	0.694	0.852
DecisionTable	0.840	0.866	0.841	0.684	0.840
DTNB	0.851	0.863	0.852	0.701	0.889
JRip	0.840	0.870	0.841	0.685	0.837
NNge	0.742	0.740	0.739	0.461	0.726
OneR	0.845	0.873	0.846	0.695	0.862
PART	0.845	0.869	0.846	0.694	0.843
Ridor	0.851	0.866	0.852	0.702	0.861
ZeroR	0.582	0.339	0.429	0.000	0.485
BFTree	0.835	0.855	0.836	0.672	0.873
DecisionStump	0.856	0.888	0.856	0.716	0.836
J48	0.845	0.869	0.846	0.694	0.847
LADTree	0.830	0.848	0.831	0.662	0.829
LMT	0.840	0.855	0.841	0.681	0.862
NBTree	0.861	0.873	0.862	0.721	0.876
RandomForest	0.840	0.855	0.841	0.681	0.854
RandomTree	0.830	0.848	0.831	0.662	0.838
REPTree	0.861	0.887	0.862	0.725	0.852
SimpleCart	0.840	0.858	0.841	0.682	0.844

Table 2: Performance classification measures for dataset1.

Secondly, in order to find out which algorithms perform best for each dataset taking several classification measures into account, we used the Iman&Davenport non-parametric statistical test [5]. This test was repeated for each of the 32 datasets and produced an ordered list of algorithms with their final rank (average rank of the 19 algorithms over the 5 performance measures), in such a way that the algorithm with the best rank (highest position in each list) is the one that performs best for the measures under consideration. According to the Iman&Davenport test, if the null-hypothesis is accepted, we state that all the algorithms are equivalent, i.e., they have a similar behavior. In contrast, if the null-hypothesis is rejected, we state that there are differences between the algorithms. For the 32 tests performed in our experiment at a significance level of $\alpha=0.1$, the null-hypothesis was rejected, thus indicating that significant differences exist between classifiers.

Therefore, in order to reveal such performance differences, a post-hoc test needs to be carried out. The Bonferroni-Dunn test [5] can be applied, since all the algorithms were compared against a control algorithm (the algorithm with the highest rank), the focus being on all the possible pairwise comparisons among them. The

critical value revealed by this test at the same significance level of $\alpha=0.1$ was 9.5331. Therefore, for each dataset, that value was added to the rank of the control algorithm, and the algorithms whose rank belongs to the interval [highest rank, highest rank + critical value] are the set of best algorithms recommended for that particular dataset, given that there are no significant differences between them.

For instance, the set of best algorithms recommended for dataset1 are shown in Table 3, in which the critical interval is [2,2+9.5331]. The remaining 10 algorithms are not recommended due to the fact that their rank is over the upper limit.

Algorithm	Ranking
NBTree	2
REPTree	2.667
DecisionStump	5
DTNB	5.25
Ridor	5.667
OneR	6.333
ConjunctiveRule	8.083
J48	8.417
PART	8.833

Table 3: Ranking of the algorithms recommended for dataset6

Finally, in order to recommend algorithms for a new dataset, we used a nearest neighbor (1-NN) approach [1]. We used the unweighted normalized Euclidean distance to find the closest dataset to the new one. In the case of categorical value (the domain feature), the distance considered was 0 in the case of matching and 1 otherwise. Then, the set of best algorithms previously calculated to the most similar dataset was recommended for the new dataset.

We carried out a second experiment to compare the results obtained when the different types of features that characterize the datasets were used. We noticed that distinct nearest neighbors were obtained for the same dataset depending on the features used. For instance, the nearest neighbors obtained for dataset1 when using different feature combinations are shown in Table 4.

		Statistic+ Complex	Statistic+ Complex+ Domain
Dataset13	Dataset11	Dataset11	Dataset22

Table 4: Nearest neighbors for dataset1 depending on the combination of used features.

As can be seen in Table 4, dataset13 (from forum domain) is the most similar to dataset1 (from report domain) when only statistics features (see Table 1) are used, but dataset11 (from forum domain) is the most similar when complex and statistics features are used together, and finally, dataset22 (from report domain) is the most similar when all the features that also take the domain into account are used (see Table 1).

Four separate tests using the hold-one-out method were directed to check which combination of features (by employing only statistical features, only complexity, both statistical and complexity, and also the domain attribute) enables the best

recommendation to be obtained. Following this hold-one-out procedure, first we calculated the similarity between each dataset and the remaining 31 datasets to select the most similar dataset. Next, the set of recommended (and previously calculated) algorithms for each dataset is considered as the real output, whereas the set of algorithms of its nearest neighbor is the predicted one. Then, several evaluation measures commonly used in pattern recognition and information retrieval systems (such as search engines and recommender systems) were computed to evaluate the quality of the recommendations. Precision and recall are the metrics employed, which are defined in terms of a set of retrieved documents in an information retrieval domain, but in this work, they are defined in terms of retrieved algorithms:

$$\text{precision} = \frac{|\{\text{real_algorithms}\} \cap \{\text{predicted_algorithms}\}|}{|\{\text{predicted_algorithms}\}|}$$

$$\text{recall} = \frac{|\{\text{real_algorithms}\} \cap \{\text{predicted_algorithms}\}|}{|\{\text{real_algorithms}\}|}$$

There is an inverse relationship between precision and recall, in such a way that obtaining higher values of one measure means obtaining lower values for the other. Nevertheless, there is another measure, called F-Measure, which combines both precision and recall and is computed as the harmonic mean of both:

$$F - \text{Measure} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

The F-measure results achieved for the four combinations of features used are shown in a box plot or box-and-whisker diagram that shows the smallest observation (sample minimum), lower quartile (Q1), median (Q2), upper quartile (Q3), and largest observation (sample maximum), as can be seen in Figure 2. As can be observed, better results are obtained when the statistical and complexity features are considered jointly rather than when they are considered independently. Moreover, the best results are reached when the domain attribute is also included.

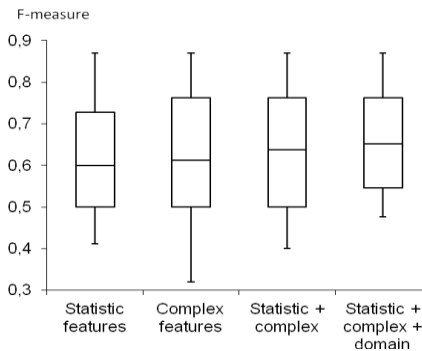


Figure 2. Blox plot of the F-measure.

5. CONCLUSIONS

In this paper, meta-learning has been used to address the problem of recommending a subset white-box classifier from Moodle datasets. Several classification performance measures are used together with several statistical test to rank and select a subset of algorithms. Results show that complexity and domain features used to characterize datasets can improve the quality of the recommendation. For future work, we plan to extend the experimentation, for example, using more datasets, algorithms (including black box models), characteristics, evaluation measures, etc.

Future research may employ a greater number of classification datasets from other sources or other kinds of education systems (primary, secondary, higher, special education, ...) in which different specific domain features to characterize datasets can be used. A further line of research would be to develop more advanced off-line procedures, such as the employment of several K-NN neighbors instead of the 1-NN, and methods for merging several rankings and subsets of algorithms in neighboring datasets.

6. ACKNOWLEDGMENTS

This work was supported by the Regional Government of Andalusia and the Spanish Ministry of Science and Technology projects, P08-TIC-3720 and TIN-2011-22408, and FEDER funds.

7. REFERENCES

- [1] Aha, D., Kibler, D. Instance-based learning algorithms. *Machine Learning*. 6, 37-66, 1991.
- [2] Asuncion, A, Newman, D.J.. UCI Machine Learning Repository, University of California, Irvine, CA, 2007. (<http://www.ics.uci.edu/mlearn/MLRepository.html>).
- [3] Bhatt, N. Thakkar, A. Ganatra, A. A Survey & Current Research Challenges in Meta Learning Approaches based on Dataset Characteristics. *International Journal of Soft Computing and Engineering*, 2(10), 234-247, 2012.
- [4] Brazdil, P., Giraud-Carrier, C., Soares, C. and Vilalta, R. *Metalearning: Applications to Data Mining. Series: Cognitive Technologies*. Springer, 2009.
- [5] Demsar, J. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7 1-30, 2006.
- [6] Hämäläinen, W., Vinni M. Classifiers for educational data mining; *Handbook of Educational Data Mining*. Chapman & Hall/CRC. 2011.
- [7] Ho T.K., Basu M. Complexity measures of supervised classification problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):289-300, 2002.
- [8] Molina, M.M., Romero, C. Luna, J.M. Ventura, S. Meta-learning approach for automatic parameter tuning: A case study with educational datasets. 5th International Conference on Educational Data Mining, Chania, Greece, 180-183, 2012.
- [9] Orriols-Puig A., Macià N. & Ho T.K. Documentation for the data complexity library in C++. Technical report, La Salle - Universitat Ramon Llull, 2010.
- [10] Romero, C. and Ventura, S. Data Mining in Education. *Wire Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 3:12-27. 2013.
- [11] Romero, C., Espejo, P., Romero, R., Ventura, S. Web usage mining for predicting final marks of students that use Moodle courses. *Computer Applications in Engineering Education*. 21(1): 135-146. 2013.
- [12] Song, Q, Wang, G, Wang, C. Automatic recommendation of classification algorithms based on dataset characteristics. *Pattern recognition*. 45, 2672-2689, 2012.
- [13] Witten, I. H., Eibe, F. and Hall, M.A. *Data Mining: Practical Machine Learning Tools and Techniques, Third Edition*. Morgan Kaufman Publishers, 2011.