Relational Ranking with Predictive Clustering Trees

Sašo Džeroski[1], Hendrik Blockeel[2], Ljupčo Todorovski[1]

[1]Jožef Stefan Institute, Jamova 39, SI-1000 Ljubljana, Slovenia
[2]Katholieke Universiteit Leuven, Celestijnenlaan 200A, B-3001 Heverlee, Belgium

In many cases, running a machine learning algorithm on a given classification task can be time consuming, especially when complex tasks are involved. It is therefore desirable to be able to predict the performance of a given algorithm on a given task from a task description (set of properties of the task) and without actually running the algorithm. Furthermore, due to the variety of machine learning algorithms for classification tasks, the problem of choosing an appropriate algorithm for a given task arises. We can view this as a multi-target prediction problem, where the same input (the task description) is used to predict several related targets (the performances of the different algorithms). We are then interested in obtaining an ordering of the algorithms (called ranking) in terms of their expected relative performance.

A typical meta-level dataset for ranking consists of two parts. The first set of columns (attributes) contain a description of the task at hand. In the case of ranking learning algorithms, it typically contains general and statistical properties of datasets (such as the number of examples and class values and the average kurtosis per numerical attribute). The second set of columns contains the performance figures for the learning algorithms on the given datasets (e.g., the classification error of C5.0, RIPPER, etc.).

A prototypical ranker uses a case-based (nearest neighbor) approach. To produce a ranking on the learning algorithms a new dataset, the most similar datasets from the meta-level dataset are chosen and the performances (rankings) of the algorithms on theses datasets are averaged to obtain a prediction of the performance (ranking) on the new dataset. In this paper, we propose to use a relational representation of tasks (datasets) in ranking instead of a propositional one. This allows us to represent the tasks in more detail, e.g., include the kurtosis values for each numerical attribute rather than include only the average kurtosis per numerical attribute. We also propose to use predictive clustering trees for ranking instead of case-based approaches. In this case, in addition to obtaining a ranking, we also obtain an explanation.

We performed experiments with both relational and propositional ranking trees on a meta-level dataset consisting of data about performance of 10 machine learning algorithms on 42 classification tasks from the UCI repository. The relational ranking trees perform better than propositional ranking trees and also better than the default ranking when a smaller amount of tree pruning is applied. As compared to existing ranking approaches which are propositional and instance-based, our approach also allows for an explanation of the predicted rankings.

An immediate direction for further work is to repeat the experimental evaluation on the full METAL ranking dataset (53 meta-level data points) once it becomes available. Given the size of the meta-level dataset, any additional point matters. This would also allow for a direct comparison to the propositional instance-based approaches to ranking.

Other directions for further work include the definition of a relational distance measure on datasets and the use of relational instance-based learning for relational ranking. Investigating the use of kernels for relational data would also be an interesting direction to pursue. Both approaches are known to work well for small datasets of high dimensionality. If we can deal with small datasets of high dimensionality, it makes sense to also consider additional dataset properties. Dataset properties based on landmarking have been shown to predict performance well and to be useful for ranking. Also, features based on the shape of decision trees induced from a datasets could be interesting in this respect.