Minimal Cost Complexity Pruning of Meta-Classifiers

Andreas L. Prodromidis and **Salvatore J. Stolfo**
Department of Computer Science
Columbia University
New York, NY 10027
{andreas,sal}@cs.columbia.edu

Integrating multiple learned classification models (classifiers) computed over large and (physically) distributed data sets has been demonstrated as an effective approach to scaling inductive learning techniques, while also boosting the accuracy of individual classifiers. These gains, however, come at the expense of an increased demand for run-time system resources. The final ensemble meta-classifier may consist of a large collection of base classifiers that require increased memory resources while also slowing down classification throughput. To classify unlabeled instances, predictions need to be generated from all base-classifiers before the meta-classifier can produce its final classification. The throughput (prediction rate) of a meta-classifier is of significant importance in real-time systems, such as in e-commerce or intrusion detection.

This extended abstract describes a *pruning* algorithm that is independent of the combining scheme and is used for discarding redundant classifiers without degrading the overall predictive performance of the pruned meta-classifier. To determine the most effective base classifiers, the algorithm takes advantage of the *minimal cost-complexity pruning* method of the CART learning algorithm (Breiman *et al.* 1984) which guarantees to find the best (with respect to misclassification cost) pruned tree of a specific size (number of terminal nodes) of an initial unpruned decision tree. An alternative pruning method using Rissanen's minimum description length is described in (Quinlan & Rivest 1989).

Minimal cost complexity pruning associates a complexity parameter with the number of terminal nodes of a decision tree. It prunes decision trees by minimizing the linear combination of the complexity (size) of the tree and its misclassification cost estimate (error rate). The degree of pruning is controlled by adjusting the weight of the complexity parameter, i.e. an increase of this weight parameter results in heavier pruning.

Pruning an arbitrary meta-classifier consists of three stages. First we construct a decision tree model (e.g. CART) of the original meta-classifier, by learning its input/output behavior. This new model (a decision tree with base classifiers as nodes) reveals and prunes the base classifiers that do not participate in the splitting criteria and hence are redundant. The next stage aims to further reduce, if necessary, the number of selected classifiers. The algorithm applies the minimal cost-complexity pruning method to reduce the size of the decision tree model and thus prune away additional base classifiers. The degree of pruning is dictated by the system requirements and is controlled by the complexity parameter. In the final stage, the pruning algorithm re-applies the original combining technique over the remaining base-classifiers (those that were not discarded during the first two phases) to compute the new final pruned meta-classifier.

To evaluate these techniques, we applied 5 inductive learning algorithms on 12 disjoint subsets of 2 data sets of real credit card transactions, provided by Chase and First Union bank. We combined (using the weighted voting and stacking (Wolpert 1992) combining schemes) the 60 base classifiers in a 6-fold cross validation manner.

The measurements show that using decision trees to prune meta-classifiers is remarkably successful. The pruned meta-classifiers computed over the Chase data retain their performance levels to 100% of the original meta-classifier even with as much as 60% of the base classifiers pruned or within 60% of the original with 90% pruning. At the same time, the pruned meta-classifiers exhibit 230% and 638% higher throughput respectively. For the First Union base classifiers, the results are even better. With 80% pruning, there is no appreciable reduction in accuracy, while with 90% pruning they are within 80% of the performance of the unpruned meta-classifier. The throughput improvements in this case is 5.08 and 9.92 times better, respectively.

References

Breiman, L.; Friedman, J. H.; Olshen, R. A.; and Stone, C. J. 1984. *Classification and Regression Trees.* Belmont, CA: Wadsworth.

Quinlan, R., and Rivest, R. 1989. Inferring decision trees using the minimum description length princliple. *Information and Computation* 80:227–248.

Wolpert, D. 1992. Stacked generalization. *Neural Networks* 5:241–259.