Experiments on solving multiclass learning problems by n^2-classifier

Jacek Jelonek and Jerzy Stefanowski

Institute of Computing Science, Poznan University of Technology,
Piotrowo 3A, 60-965 Poznan, Poland
jacek.jelonek@cs.put.poznan.pl
jerzy.stefanowski@cs.put.poznan.pl

Abstract. The paper presents an experimental study of solving multiclass learning problems by a method called n^2-classifier. This approach is based on training $(n^2 - n)/2$ binary classifiers - one for each pair of classes. Final decision is obtained by a weighted majority voting rule. The aim of the computational experiment is to examine the influence of the choice of a learning algorithm on a classification performance of the n^2-classifier. Three different algorithms are considered: decision trees, neural networks and instance based learning algorithm.

1 Introduction

In this paper, we focus our attention on using multiple classifiers to solve *multiclass learning problems*. The multiclass learning problem involves finding a classification system that maps descriptions of training examples into a discrete set of n decision classes ($n > 2$). Although the standard way to solve multiclass learning problems includes the direct use of the multiclass learning algorithm such as, e.g. algorithm for inducing decision trees, neural network, or instance-based algorithm, there exist more specialized methods dedicated to this problem. As it is discussed in literature such approaches, e.g., one-per-class method, distributed output codes classification schemes, error-correcting techniques (ECOC) can outperform the direct use of the single multiclass learning algorithms (see, e.g. [3, 4, 8, 10]).

We consider another model which we called the n^2-classifier. It is inspired by the concept of multiple classification models [3]. The n^2-classifier is composed of $(n^2 - n)/2$ base binary classifiers. Each base classifier is specialized to discriminate respective pair of decision classes. A new example is classified by applying its description to all base classifiers. Then, their predictions are aggregated to a final classification decision using a weighted majority voting rule.

This approach is quite similar to the concept of *pairwise coupling* classification which was independently introduced in [5, 6]. Our n^2-classifier approach differs, however, from the above concept by using another combination rule. It takes into account the information about a class that is indicated by majority of base classifiers. Additionally, the voting scheme is adjusted by the credibility of the base classifiers, which are calculated during learning phase of classification.

As it has been indicated in [5, 6, 7] such integration of binary classifiers performs usually better than the respective, single multiclass classification model. One of the important aspects of constructing the homogenous n^2-classifier is the choice of learning algorithms to be used by base classifiers. We think that the expected improvement of classification accuracy may depend on both the particular problem and used proper base classifier.

Therefore, the main research aim of the following study is to perform an evaluation of the homogeneous n^2-classifier constructed by various base classifiers. Several known learning algorithms may be employed. However, we think that algorithms with inherent capability of reducing the influence of irrelevant features could be more appropriate in this approach than algorithms in which all features are treated as equally important. According to this hypothesis we decided to compare usefulness of three different learning algorithms, i.e. decision trees, neural networks and instance based learning.

2 The n^2-classifier

The n^2-classifier belong to the group of multiple classification models adopted to solve multiclass learning problems. The main principle of the n^2-classifier is the discrimination of each pair of the classes: (i, j); $i, j \in [1..n]$ $i \neq j$, by an independent binary classifier C_{ij}. The classifier C_{ij} produces a binary classification indicating whether a new example $\mathbf{x}$ belongs to class i or to class j. Let $C_{ij}(\mathbf{x})$ denotes the classification of an example $\mathbf{x}$ by the base classifier C_{ij}. We assume that $C_{ij}(\mathbf{x}) = 1$ means that example $\mathbf{x}$ is classified by C_{ij} to class i, otherwise ($C_{ij}(\mathbf{x})$= 0) $\mathbf{x}$ is classified to class j. Based on definition: $C_{ij}(\mathbf{x})$ = 1 - $C_{ji}(\mathbf{x})$.

For a new example $\mathbf{x}$, a final classification is obtained by an aggregation of the base classifiers predictions - $C_{ij}(\mathbf{x})$. The simplest aggregation is based on finding a class that wins the most pairwise comparisons. The classification performance of base classifiers is usually diverse because they are trained on different pairs of classes. So, it is necessary to estimate their credibility. In this study we assume that with each classifier C_{ij} we associate a credibility coefficient P_{ij} defined in following way:

$$P_{ij} = \frac{v_i}{v_i + e_j}$$

where e_j is a number of misclassified examples from class j, and v_i is a number of correctly classified examples from class i. The computation of the credibility coefficients is performed during the learning phase of constructing the n^2-classifier (i.e. done on the training examples). Final classification decision is determined by a weighted majority voting rule, which indicates to choose such a decision class i for which the following formula returns the maximum value:

$$\sum_{j=1, i \neq j}^{n} P_{ij} \cdot C_{ij}(\mathbf{x})$$

The introduced definition of the n^2-classifier is general and therefore any base learning algorithm can be employed in this framework.

3 Computational experiments

We performed learning decision trees using our own implementation based on a Quinlan's ID3 algorithm. This implementation contains some of the modifications introduced in the Assistant system [2], i.e. binarization and prepruning of decision trees. Artificial neural networks were implemented as typical feed forward multi-layer networks. The instance based learning algorithm is a typical approach based on k nearest neighbor principle [1]. We implemented a non-incremental version of IBL1, where all training examples are stored.

Table 1. Data sets used in the experiments

No.	Data set	Number of examples	Number of classes	Number of attributes
1.	Automobile	159*	6	25
2.	Cooc	700	14	22
3.	Ecoli	336	8	7
4.	Glass	214	6	9
5.	Hist	700	14	17
6.	Meta-data	528	5*	20
7.	Primary Tumor	339	21	17
8.	Soybean-large	542*	14*	35
9.	Vowel	990	11	10
10.	Yeast	1484	10	8

All computation experiments have been performed on the typical benchmark data sets. Some characteristics of the employed multiclass data sets are summarized in Table 1. The most of them are coming from the Machine Learning Repository at the University of California at Irvine [9]. The Cooc and Hist data sets come from our previous experiments and concern the recognition of tumors of the central nervous system on the basis of features extracted from microscopic images. Some of the studied data sets have been slightly modified - what is indicated in Table 1 by asterisks. First modifications concern the choice of decision attributes for two problems, i.e. for Automobile data set we have used the first ("symboling") attributes, and the Meta-data set is characterized by continuos decision attribute which has been discretized using thresholds: 6, 13, 20 and 50, thus giving five classes. Then, for Automobil and Soybean-large data sets we removed examples or attributes containing too many missing values. In the case of the Meta-data and the Primary Tumor, missing values have been replaced by the most frequent values. The classification accuracy was estimated by stratified version of 10-fold cross-validation technique, i.e. the training examples were partitioned into 10 equal-sized blocks with similar class distributions as in the original set.

Table 2. Performance of n^2-classifier based on decision tree (n^2_{DT}) and single decision tree (DT)

No.	Name of data set	Accuracy of DT (%)	Accuracy of n^2_{DT} (%)	Improvement n^2 vs DT (%)
1.	Automobile	85.5 ± 1.9	87.0 ± 1.9	1.5* ± 1.8
2.	Cooc	54.0 ± 2.0	59.0 ± 1.7	5.0 ± 1.0
3.	Ecoli	79.7 ± 0.8	81.0 ± 1.7	1.3 ± 0.7
4.	Glass	70.7 ± 2.1	74.0 ± 1.1	3.3 ± 1.8
5.	Hist	71.3 ± 2.3	73.0 ± 1.8	1.7 ± 1.7
6.	Meta-data	47.2 ± 1.4	49.8 ± 1.4	2.6 ± 1.3
7.	Primary Tumor	40.2 ± 1.5	45.1 ± 1.2	4.9 ± 1.5
8.	Soybean-large	91.9 ± 0.7	92.4 ± 0.5	0.5* ± 0.7
9.	Vowel	81.1 ± 1.1	83.7 ± 0.5	2.6 ± 0.7
10.	Yeast	49.1 ± 2.1	52.8 ± 1.8	3.7 ± 2.2

Table 3. Performance of n^2-classifier based on neural network (n^2_{ANN}) and single artificial neural network (ANN)

No.	Name of data set	Accuracy of ANN (%)	Accuracy of n^2_{ANN} (%)	Improvement n^2 vs ANN (%)
1.	Automobile	52.6 ± 2.0	58.1 ± 2.3	5.5 ± 1.1
2.	Cooc	56.0 ± 1.9	65.3 ± 0.7	9.3 ± 1.4
3.	Ecoli	81.7 ± 1.7	83.0 ± 1.6	1.3* ± 2.0
4.	Glass	62.7 ± 2.0	62.8 ± 0.8	0.1* ± 1.6
5.	Hist	65.7 ± 3.0	83.3 ± 1.4	17.6 ± 2.0
6.	Meta-data	50.5 ± 1.6	47.2 ± 1.5	-3.3 ± 1.2
7.	Primary Tumor	38.2 ± 1.5	43.4 ± 1.2	5.2 ± 1.5
8.	Soybean-large	90.1 ± 0.8	92.9 ± 0.7	2.8 ± 0.7
9.	Vowel	59.7 ± 2.4	86.1 ± 1.0	26.4 ± 2.3
10.	Yeast	53.1 ± 1.4	59.0 ± 0.9	5.9 ± 1.0

Table 4. Performance of n^2-classifier based on IBL algorithm (n^2_{IBL}) and single instance based learning algorithm (IBL)

No.	Name of data set	Accuracy of IBL (%)	Accuracy of n^2_{IBL} (%)	Improvement n^2 vs IBL (%)
1.	Automobile	77.7 ± 0.9	76.7 ± 1.0	-1.0 ± 0.2
2.	Cooc	68.4 ± 0.6	68.3 ± 0.6	-0.1 ± 0.1
3.	Ecoli	81.3 ± 0.5	81.3 ± 0.4	0.0* ± 0.2
4.	Glass	68.8 ± 0.8	68.5 ± 1.0	-0.3* ± 0.5
5.	Hist	89.3 ± 0.5	89.3 ± 0.5	0.0 N/A
6.	Meta-data	40.6 ± 1.6	42.1 ± 1.6	1.5 ± 0.6
7.	Primary Tumor	33.4 ± 1.2	36.2 ± 1.5	2.8 ± 1.2
8.	Soybean-large	89.9 ± 0.4	89.9 ± 0.4	0.0 N/A
9.	Vowel	98.9 ± 0.2	98.9 ± 0.2	0.0 N/A
10.	Yeast	52.8 ± 0.7	53.3 ± 0.7	0.5 ± 0.2

The validation technique was repeated 10 times for each data set. For each average accuracy we calculated the standard deviation. The improvement of n^2-classifier is expressed as the difference of average accuracy of the appropriate classifiers with a confidence interval. It was calculated based on a t-test for paired differences of means, with confidence level 0.95. An asterisk indicates that the difference of the accuracy is not statistically significant.

First, we evaluated the classification performance of the n^2-classifier based on decision trees. We also compared it to the single multiclass decision tree (DT). All decision tree classifiers were trained in a unpruned manner. The results of the experiment are presented in Table 2.

Then, we tested the performance of the n^2-classifier employing artificial neural networks. We systematically checked various topologies of networks depending on the particular data, e.g. for data sets with smaller number of input features (ecoli, glass, vowel, yeast) we tested the following number of neurons in input and hidden layers: 8, 10, 12, 14. Moreover with each combination of these topologies we tested various number of epoch: 50, 100, 150, 250. It means that for each learning problem we systematically looked through 64 combinations to find the best learning parameters. The results of the experiments with n^2-classifier and single classification model (ANN) for neural networks are presented in Table 3.

As the third classification model, we examined instance based learning algorithm. The computation results are presented in Table 4 in an identical way as in previous tables.

4 Conclusions

Let us summarize the results obtained for the particular learning algorithms. In a case of applying the decision tree as a base classifier we can observe that in 8 of all (10) problems the integration of decision trees into the n^2-classifier results in significantly better classification accuracy than the direct use of multiclass single decision tree. For two remaining problems the improvement is indistinguishable. The highest improvement is observed for Cooc data set - 5.0%. Similarly for neural networks the results show that the n^2-classifier performs generally better than single multiclass approach. The increase of classification accuracy is noticed in 9 of 10 data sets. Moreover, the improvements are relatively higher than for decision trees. Particularly high increase is observed for Vowel data - 26.3%. On contrary using IBL usually does not result in better classification ability of the n^2-classifier. The increase exists only for 3 data sets. For the remaining ones the results are similar, while for two data sets the classification ability slightly decreases for the n^2-classifier.

The obtained results showed clearly that the classification performance of the introduced n^2-classifier is generally better than the accuracy of single classifier approach for two considered base learning algorithms, i.e. decision trees and neural networks. Let us also notice that experimental results presented in [5, 6] also indicate that coupling strategy improves the classification accuracy although the relative performance of different approaches depends on the problem.

In our case study, we can summarize that the neural network seems to be the best model for the n^2-classifier. The decision trees are the second model according to the improvement of the classification accuracy. On the other hand the use of instance based learning algorithm is not so encouraging. Its the worst performance could result from the fact that IBL treats all features as equally important while two former approaches have inherent capability of reducing the irrelevant features what may help with defining proper subspace of features for efficient solving two-class problem.

There exist several on-going research problems that could be investigated in the future within the n^2-classifier framework. For instance, one can analyze the problem using the architecture of heterogeneous base classifiers or verify an idea of using n^2-classifier in constructive induction problems.

Acknowledgements
The computational experiments have been partially carried out at the Poznan Supercomputing and Networking Center affiliated to the Institute of Bioorganic Chemistry at the Polish Academy of Sciences. Research on this paper was supported by the grant KBN no. 8T11C 013 13 and CRIT 2 - Esprit Project no. 20288.

References

1. Aha D.W., Kibler E., Albert M.K.: Instance-based learning algorithms. Machine Learning, **6**, (1991) 37-66.
2. Cestnik, B., Kononenko, I., Bratko, I.: Assistant 86, a knowledge elicitation tool for sophisticated users. In Bratko I., Lavrac N. (eds.) Progress in Machine Learning, Sigma Press, Wilmshow, (1987) 31-45.
3. Chan, P.K., Stolfo, S.J.: Experiments on multistrategy learning by meta- learning. In Proceedings of the Second International Conference on Information and Knowledge Management, (1993) 314-323.
4. Dietterich, T.G., Bakiri, G.: Solving muliclass learning problems via error- correcting output codes. Journal of Artificial Intelligence Research, **2**, (1995) 263-286.
5. Friedman, J.H.: Another approach to polychotomous classification, Technical Report, Stanford University, 1996.
6. Hastie, T., Tibshirani R.: Classification by pairwise coupling, Proc. NIPS97.
7. Jelonek, J., Stefanowski J.: Using n^2-classifier to solve multiclass learning problems. Technical Report, Poznan University of Techonology 1997.
8. Mayoraz, E., Moreira, M.: On the decomposition of polychotomies into dichotomies, Proc. 14th Int. Conf. Machine Learning, July 1997, 219-226.
9. Murphy, P.M., Aha, D.W.: Repository of Machine Learning. University of California at Irvine. [URL: http://www.ics.uci.edu/ mlearn/MLRepositoru.html].
10. Schapire, R.E. Using output codes to boost multiclass learning problems. In Proceedings of the 14th International Machine Learning Conference (1997).

This article was processed using the LaTeX macro package with LLNCS style