Annales UMCS Informatica AI 5 (2006) 145-153

Annales UMCS
Informatica
Lublin-Polonia
Sectio AI

http://www.annales.umcs.lublin.pl/

Reduction of ensemble of classifiers with a rule sets analysis

Ewa Szpunar-Huk*

Faculty of Computer Science and Management, Wrocław University of Technology,
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland

Abstract

The article shortly discusses the aim of classification task and its application to different domains of life. The idea of ensemble of classifiers is presented and some aspects of grouping methods are discussed. The paper points to the need of ensemble classifier pruning and presents a new approach for ensemble reduction. The proposed method is dedicated to committees of decision trees and bases on transformation of a tree set into a rule set and the new, suited to the pruning method, the weighted voting algorithm is also presented. There are also described experiments showing properties and effectiveness of the proposed method. Finally, directions of further research are mentioned.

1. Introduction

The amount of data stored in databases continues to grow rapidly. This large amount of data contains potentially valuable hidden knowledge. Data mining is the core step of a broader process, called knowledge discovery in databases. Data mining systems aim to discover patterns and extract useful information from facts recorded in databases. Acquiring knowledge from databases is to apply various machine learning algorithms to compute descriptive representation of the data as well as patterns that may be exhibited in the data. One of the most studied tasks of data mining is a classification task, whose aim is to run a learning algorithm on a set of training examples, to produce a classifier. The classifier is a data model, which given a new example, predicts the corresponding class label called a classifier, which enables automated data labeling. The aim of the classification task is to train a classifier that minimizes the error in predictions on an independent test set of examples (generalization error).

Classifiers were applied to variety of problems from strictly scientific investigation to practical ones, like making improvements to a decision-making process of an organization. As an example of successful research, which lead to

* *Email address*: ewa.szpunar@pwr.wroc.pl

new scientific discovery using classifiers, can be mentioned The Optical Gravitational Lensing Experiment (OGLE) – a project with the main goal of searching for the dark matter with microlensing phenomena, and The All Sky Automated Survey (ASAS) – a project which allowed to build a Catalogue of Variable Stars. The initial idea for both projects is due to prof. Bohdan Paczyński and exhaustive information can be found at a home page of the Warsaw University Astronomical Observatory (http://www.astrouw.edu.pl/).

It is not easy to find the area, where automatic classification can not be useful. Often it is used to predict outcomes for future situations as an aid to decision-making process, i.e. identifying more accurate models in financial decision domains including credit scoring and bankruptcy prediction [1]. Widely studied are also diagnostic decision support systems, like i.e. breast cancer diagnosis applications [2]. Automated image analysis is popular as well, i.e. classifying satellite images to obtain a catalog of land cover types [3]. In ecological modeling models like spatial simulations of fire and vegetation dynamics can be classified [4]. Even in typical predicting systems classifier can be useful [5]. Such examples can be multiplied, but it is not possible to mention all of them.

2. Ensemble of classifiers

The information to design a classifier is usually in the form of labeled data $D = \{t_1,t_2,..,t_N\}$, where $t_i = <\underline{x},c> \in X \times C$ from set D is a data vector $\underline{x} = <x_1,x_2,..,x_m,>$ associated with a class label c from a discrete set of classes C, X_i is a domain of x_i and $X = X_1 \times X_2 \times \ldots \times X_m$ is a domain of data vector. Components of the vector $\underline{x}$ are usually real or discrete (nominal) values, such as height, weight, age, eye-color, and so on. These components of the data vector are often referred to as the features or attributes of an example. A classifier is, for a given data set, a hypothesis about the approximated function $F: X \rightarrow C$.

Examples of simple classifiers are: linear and quadratic discriminants, the k-nearest neighbor rule, the Parzen density classifier, decision tree and neural network. Some classifiers are very flexible, with many user adjustable parameters – others are almost entirely automatic. Some of these classifiers are metric dependent. It has therefore to be assumed that the metric is given. In the case of decision trees the problem is the size of the tree. If a tree is grown until its natural end of zero error, it is oversized (over-trained) and might perform badly on an independent test set. Pruning or early stopping is necessary. On the other hand, neural networks are very problematic for designing clear automatic classifiers. There is no single "best" classifier. Classifiers applied to different problems perform differently [6].

One of the major advances in inductive learning in the last 15 years was the development of ensemble approaches [7]. Although there are many unanswered

questions about matching classifiers to real-life problems, combining classifiers is a rapidly growing research area. An ensemble of classifiers is a set of classifiers whose individual decisions are combined in some way (typically by voting) to classify new examples. This area is referred to by different names in literature – committees, mixtures of experts, classifier ensembles, multiple classifier systems, consensus theory, etc. [6]. In general, an ensemble method is used to improve the accuracy of a given classifier. In this paper a simple classifier is referred to as the base classifier.

Combining methods become popular, because of the fact that, although for many data set it is easy to find hypothesis, which exactly classifies training set, it is unusual to find a single expert achieving the best results on the overall problem domain. Experimental results showed that ensemble methods can significantly increase prediction abilities of the whole system. The need of general theory that underpins classifier combination has been acknowledged regularly, but such theory does not exist as yet [6]. There are plenty of methods developed for the ensembles construction but they have to respect the assumption, that base classifiers in an ensemble should be different from each other, otherwise there is no gain in combining them. The base classifiers can be of any type mentioned above, different data subsets and different feature subsets can be used to build base classifiers, and finally there can be different methods the classifier ensemble is making decision. The architecture of an ensemble system and four approaches aiming at building ensembles of diverse classifiers are graphically illustrated in Figure 1.

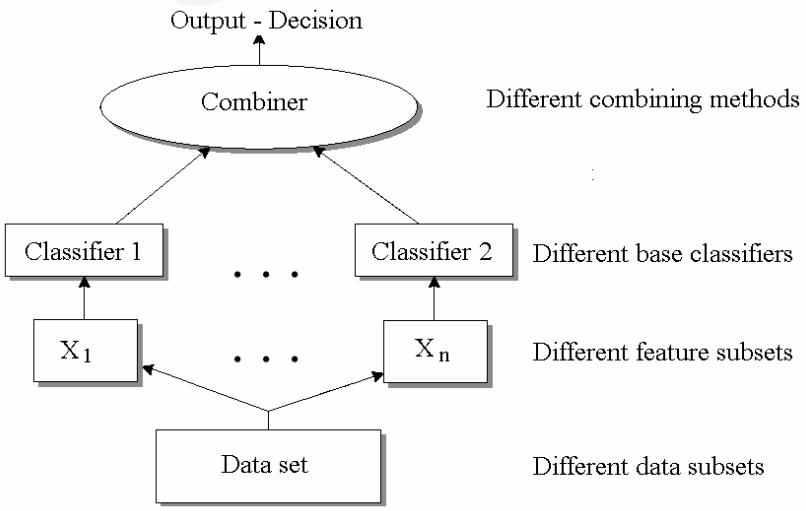

Fig. 1. The architecture of an ensemble system

But there are also some drawbacks. In general, ensemble classifiers have an internal representation that cannot be easily viewed or parsed. This can discredit ensemble in some application areas, where justification or presentation of decision making process is essential, like in medical diagnostic systems. Besides, complex classifiers demand more resources to be stored and a longer time to make a decision.

3. Ensemble pruning

Ensemble pruning methods consist in producing a pool of classifiers followed by the selection procedure to pick the base classifiers to make the whole ensemble more diverse and accurate. Such methods are also called reduction, thinning the ensemble or overproduce and select. They can reduce overfitting or time for classification. Most of these methods base on specifying characteristics, so called diversity measure, and choosing the most diverse base classifiers. There is no strict definition of what is intuitively perceived as diversity, but it is considered as a measure of probability, that base classifiers do not make coincident errors [6]. Methods for building ensembles, which rely on inducing diversity in an intuitive manner, are very successful [8]. Example of such methods can be: the Q statistic [9] or Percentage Correct Diversity Measure (PCMD) [10]. The common property of these methods is that during a comparison of base classifiers they only consider their performance on a data set and do not take into consideration structure of classifiers. In contrast to these approaches, the reduction method proposed in this paper, is based on an internal structure of base classifiers.

The proposed method is designed for ensemble classifiers, which are made up of classification trees as base classifiers. Decision trees offer some advantages, because they can handle both continuous and nominal data, generate interpretable classification rules, are fast to train and often are as accurate as or even slightly more accurate than many other classifiers. In the proposed approach a simple tree is treated as a set of Horn's rules, which make it possible to transform a set of classification trees into a single rule set. During the transformation process duplicated rules are removed which is equivalent to the elimination of covering parts of trees. It allows to modify a method the decision of an ensemble is made, by giving a voting right also to those rules, for which more than a given percent of conditions (PC) for a particular data vector are satisfied. Finally, weight of a vote of individual rules depends on a percent of satisfied conditions, but what is more, rules with less than 100% satisfied conditions have a vote's weight additionally reduced. This reduction depends on a parametrically given factor (WR). For completeness, a detailed description of the training and classification phase for the pruning algorithm is provided in Figure 2.

Input data:
1. $D = \{t_1,t_2,..,t_N\}$, where $t_i = <\underline{x},c>$
2. $c_1,..,cj$ – class labels
3. $E = \phi$ the ensemble of classification trees
4. $RE = \phi$ the ensemble of classification trees reduced to a rule set

Training phase:
1. Train the ensemble of trees
2. For each tree $T \in E$ do
   - Turn T into a rule set R_T
   - For each rule $r \in R_T$ do
     - If $r \notin R$ then add r to RE

Classification phase:
1. Initialize the parameters
   - PC – percent of satisfied conditions
   - WR – vote's weight reduction coefficient
   - For each class label c_i set the number of votes Vc_i to 0
2. For each rule $r \in RE$ do
   - Run r on the input vector $\underline{x}$ and count p – a percent of satisfied conditions
   - If $p \geq PC$ then
     - Count the weight of a voice $w = WR*p$
     - Increase proper Vc_i value by w
3. The class c_i with the maximum value of Vc_i is chosen as the label for $\underline{x}$

Fig. 2. The training and classification phase for the proposed pruning method

4. Experimental results

This section is devoted to the empirical evaluation of the proposed method. Experiments were performed on 10 publicly available data sets form UCI Machine Learning Repository [11]. Each data set was divided into two subsets: training and testing set If a data set was not originally divided, it was split in the ratio 90% (training set) to 10% (testing set). The experiments mainly aimed at investigating the classification error of ensemble on a training set before and after application of a proposed reduction method. The error here is counted as the percent of misclassified objects. Each experiment was repeated 100 times.

Firstly, an ensemble of classification trees was built and the performance of ensemble before reduction d(E) was analyzed. As a learner of base classifiers, C4.5 algorithm was used – one of the best algorithms for generating classification trees [12]. Additionally bagging was used as a method for generating diverse ensemble. Breiman introduced the term bagging as an acronym for Bootstrap AGGregatING [13]. The idea of bagging is simple and

appealing: the diversity necessary to make the ensemble work is created by using different training sets – each classifier is trained on a set of N training examples, drawn randomly with replacement from the original training set of size N. Such a training set is called a *bootstrap replicate* of the original set. Bagging was compared by Banfield [14] against 7 most popular ensemble of trees creation techniques, and none of the methods he considered appeared generally, statistically, significantly more accurate than bagging.

After the ensemble has been generated, it was reduced and its error *d(RE)* was counted. Table 1 shows the averaged classification results obtained during 100 runs for an ensemble of 40 decision trees: both error values *d(E)* and *d(RE)*, percentage error changes and percent of rules removed during the reduction. For comparison, the results obtained for only one classification tree are also included. The reduction parameters used during these experiments were: *WR* = 0.1 and *PC* = 80.

Table 1. Averaged classification errors for the ensembles of 40 classification trees for 100 runs of algorithm with WR = 0.1 and PC = 80

Data set name	Decision tree error	Error before reduction $d(E)$	Error after reduction $d(RE)$	Errors reduction $d(E)$-$d(RE)$	Percentage errors reduction $(d(E)-d(RE))*100/d(E)$	Percent of removed rules
Balance	32.56	22.55	21.35	1.20	5.32	39.83
Bupa	37.53	29.06	28.65	0.41	1.41	12.53
Heart	22.80	20.39	19.68	0.71	3.48	23.9
Glass	35.33	31.62	33.05	-1.43	-4.52	12.54
Ionosphere	17.43	18.00	16.29	1.71	9.50	27.67
Pima-indians-diabetes	19.37	13.24	12.25	0.99	7.47	12.5
Sonar	9.82	0.25	0.16	0.08	34.60	5.89
Soybean	12.77	7.54	6.80	0.73	9.73	22.29
Vote	4.14	2.61	2.32	0.29	11.31	59.46
Wine	3.76	0.43	0.33	0.10	23.25	44.97

The conducted experiments illustrate and confirm what has already been found by others [6,7,13], that using ensemble classifiers can significantly reduce classification error compared to a single classifier. But before analyzing a performance of a pruning method, it is worth to look into the influence of main parameters of the algorithm for the changes in error reduction. Thus next experimental results show, how the level of error reduction for the same data set changes, when the values of *PC* and *WR* are modified. Table 2 shows the average percentage errors reduction obtained during 100 experiments for 4 selected data sets, ensembles of 40 classification trees, parameter *WR* = 80 and

different values of *PC* parameter. In Table 3 the results of similar experiments are shown, but the value of *WR* is set to 0.1 and *PC* parameter changes.

Table 2. Average percentage errors reduction for an ensemble of 40 classification trees for 100 runs of algorithm with WR = 0.1

	Percent of conditions (PC) [%]						
Data set name	40	50	60	70	80	90	100
Wine	32.20	55.56	38.10	22.00	21.73	41.11	33.33
Vote	7.64	7.46	6.54	11.11	11.31	9.65	7.69
Heart	6.10	9.74	10.71	5.56	3.48	-7.49	-2.89
Soybean	-380.05	-330.52	-83.60	-21.84	9.73	-0.66	-1.18

Table 3. Average percentage errors reduction for an ensemble of 40 classification trees for 100 runs of algorithm with *PC* = 80

	Weight reduction coefficient (*WR*)						
Data set name	0.05	0.1	0.15	0.2	0.35	0.45	0.55
Wine	48.28	23.26	19.15	50.00	46.94	47.46	54.55
Vote	10.45	11.31	10.29	8.75	7.89	9.64	0.38
Heart	-8.43	3.48	2.32	4.37	7.65	5.15	10.34
Soybean	5.44	9.73	5.15	2.45	0.77	-18.10	-26.70

The experiments showed that although one of the best methods – bagging, was used to generate ensembles, for most data sets percent of removed rules was quite big. This shows, how difficult it is to generate both diverse and accurate base classifiers. The smallest percentage reduction of the number of rules was for a data set named *Sonar*. This is in a way predictable since, as this data set contains only continuous values. For the comparison, the data set *Vote*, for which reduction was almost 60% contains only discrete values and classification tree generated with C4.5 can not be in an easy way diverse, which lies in specificity of decision tree structure. But there is no straight relationship between types of values and amount of removed rules.

For only one data set, percentage error reduction was negative, which means that for the parameters the experiments were conducted, the proposed reduction method did not reduce the classification error of an ensemble of classifiers. For some data sets, like *Sonar* and *Wine* the error was significantly reduced. But the experiments conducted to analyze influence of parameter settings showed their important role. Taking into consideration only rules with all conditions satisfied or establishing the great *WR* value does not give such advantage as otherwise. But the optimal parameter values significantly differ among data sets. The results described in Table 1. are not always the best possible to achieve, so there is need for an algorithm, which will be able to set optimal value of *PC* and *WR*, and thus tune the method to a particular problem. It could potentially depend on

the average tree height or on level of its classification error etc., but it is still an open question.

Selection of PC and WR parameter values can be realized by searching of such values, for which the described classification mechanism achieves the best generalization level on the training data. The preliminary results show that such choice secures also significant reduction of classification error level on the test data. Research on modification of this method is being currently conducted, which is hoped to allow selection of PC and WR parameter values, optimal for the described method of classification with the use of rule sets.

Finally, it is worth mentioning that the maximum computational complexity of the proposed method for reduction of an ensemble to a set of rules in Landau's notation is $O(N^2)$, where N is the total number of branches of all trees included in the ensemble. This is in the case of every branch different from each other. Time complexity of voting method is O(M), where M is number of rules in the final set of rules.

Conclusions

In this paper a new interesting method for reduction of an ensemble of classifiers was proposed. This method allows for significant improvement generalization abilities of the ensemble. It was shown that taking into consideration an internal structure of base classifiers can be an efficient way of ensemble reduction. The conducted experiments confirmed the predicted properties of the method. However, to have maximum profit on transformation of a tree set to one rule set with a special voting algorithm an automated tuning algorithm is needed. At the moment it promotes further research directions.

Acknowledgement

The above research is financed by the Polish Ministry of Education and Scientific Research within the 3 T11C 05729 research grant.

References

[1] West D., Dellana S., Oian J., *Neural network ensemble strategies for financial decision applications*, Computers and Operation Research, Oxford, 32 (2005).

[2] West D., Mangiameli P., Rampal R., West V., *Ensemble strategies for a medical diagnostic decision support system: A breast cancer diagnosis application David Westa*, European Journal of Operational Research, 162 (2005).

[3] Homer C.G., Huang C.,Yang L., Wylie B., *Development of a Circa 2000 Landcover Database for the United States*. ASPRS Proceedings, Washington D.C., 2002

[4] Keane R.E., Caret G.J., Davies I.D., Flannigan M.D., Gardner R.H., Lavorel S., Lenihan J.M., Li C., Scott Rupp T., *A classification of landscape fire succession models: spatial simulations of fire and vegetation dynamics*, Ecological Modelling, 179 (2004).

[5] Hadjimichael M., Bankert R.L., Kuciauskas A.P., Richardson K.A., Vogel G.N, A*pplication of knowledge discovery from databases to remote weather assessment*, Proceedings,

3rd Conference on Artificial Intelligence Applications to the Environmental Science, Long Beach CA, (2003).
[6] Kuncheva L.I., *Combining Pattern Classifiers. Methods and Algorithms*, Wiley, (2004).
[7] Dietterich T.G., *Ensemble methods in machine learning*, Multiple Classifier Systems. First International Workshop, MCS 2000, Cagliari, (2000).
[8] Kuncheva L.I., Whitaker C.J., *Measures of diversity in classifier ensembles*, Machine Learning, 51 (2003).
[9] Wei F., Fang C., Haixun W., Philips S.Y., *Pruning and Dynamic Scheduling of Cost-Sensitive Ensembles*, AAAI, (2002).
[10] Banfield R.E., Hall L.O., Bowyer K.W., Kegelmeyer W.P., *A New Ensemble Diversity Measure Applied to Thinning Ensembles*, International Workshop on Multiple Classifier Systems, Surrey, (2003).
[11] UCI Machine Learning Repository, http://www.ics.uci.edu/~mlearn/MLRepository.html
[12] Quinlan J.R., *C4.5 Programs for Machine Learning*, Morgan Kauffman, (1993).
[13] Breiman L., *Bagging predictors*. Technical Report 421, Department of Statistics, University of California, Berkeley, (1994).
[14] Banfield R. E., Hall L.O., Bowyer K.W., Bhadoria D., Kegelmeyer W.P., Eschrich S., *A Comparison of Ensemble Creation Techniques*, The Fifth International Conference on Multiple Classifiers systems, Cagliari, (2004).