A novel multiple classifiers integration algorithm with pruning function

Min Fang
Institute of Computer Science, Xidian University Xi'an, China, 710071
mfang@mail.xidian.edu.cn

Abstract

For improving identification rate and real time of ensembles learning algorithm, the diversity of ensemble classifiers is analyzed and a novel combination algorithm with pruning function of multiple classifiers is presented. A coincident errors measure of classifiers is presented for the compound error probability by which classifiers are partitioned, and some classifiers in a partition are pruned. The voting weights of pruned classifiers are assigned according to diversity between classifiers, so that optimize classifier set and voting weights for integration are obtained. The UCI data depository and Radar Radiant Point data are used as test data, and the result of experiment show that classifiers ensemble with pruning can get similar classification accuracy as accuracy of entire classifier integration and reduce classification time.

1. Foreword

Multiple classifiers ensemble method demands classifiers participating in ensemble should be accurate and diversity. It means that classification errors of the classifiers should be independent in order to improve the ensemble classification accuracy. It has been proved by Hansen, etc that when the individual neural networks participating in the integration are accurate and have a larger diversity, the effect of ensemble is better [1]. Classification accuracy can be well enhanced when the compound classifiers produce independent errors. Tumer[2] has pointed out that the enhancement of classification accuracy compared to the fusion method depends more on error incorrelation in the classification.

In the application of practical problems, it's hard to train classifiers that have independent errors. It often needs a large number of ensemble classifiers to achieve satisfactory classification accuracy. AdaBoost Ensembles Learning Algorithm has a great effect for producing classifier sets that have diversity. But it will produce a larger number of classifier sets to achieve lower forecast classification errors. This not only requires a large amount of memory to store classifiers, but also increases recognition time of classification [3]. For example, in Character Recognition System, we use AdaBoost to produce 200 base classifiers of C4.5 and achieve good classification accuracy. But are these 200 classifiers necessary? Dragos D.Margineantu and Lavarevic, A. use Kappa Measure to scale the diversity measure between classifiers [4][5]. They use Kappa measure on disagreement of two classifiers as the measure of distance function, calculate the Kappa measure of each classifier produced by AdaBoost algorithm and accomplish the cluster analysis of the classification output produced by classifiers. The lower the Kappa measure is, the higher the ensemble diversity of classifiers is. The classifiers combination having the lowest Kappa measure is chosen until reaching the scheduled the number of classifiers. Because the algorithm uses output of each classifier for all training samples as vectors to cluster, when there are more training samples, the dimension of vector will be larger. It will have a larger computational complexity.

Therefore, measure function based on the compound error probability of classifiers is proposed according to the multiple classifiers diversity, and the pruning method of ensemble classifiers based on the compound error probability is studied in this paper. The voting weights of pruned classifiers are assigned, so that optimize classifier sets and voting weights for integration are obtained. The UCI data depository and Radar Radiant Point data are used as test data, and the result of experiment show that classifiers ensemble with pruning can get similar classification accuracy as accuracy of entire classifier integration.

Supported by the YF07012 of Xi'an science innovation projects

2. Diversity analysis of ensemble classifiers

In multiple classifiers ensemble, what is the relationship between the number and the diversity of classifiers, and whether the diversity of classifiers is increased with the number of classifiers growing? Because diversity analysis is crucial for designing of

978-0-7695-3305-6/08 $25.00 © 2008 IEEE
DOI 10.1109/FSKD.2008.398

base classifiers in ensemble and studying of ensemble technology, problems like how to define the diversity between multiple classifiers and so on have become the key issues of this field which need to be solved urgently [6].

2.1 Plain disagreement measure

Plain disagreement measure is a commonly used method which uses the diversity of classification behavior between classifiers to provide the diversity measure. G.Zenobi uses plain disagreement measure in fitness function [7] to guide to produce ensemble classifiers. For classifiers i and j, plain disagreement measure is defined as

$$D\_plain(i,j)=\frac{1}{N}\sum_{k=1}^{N}Diff(h_i(x_k),h_j(x_k)) \qquad (1)$$

where N is the number of samples of the data set. $h_i(x_k)$ is the class label which is designated for sample x_k by classifier i. Function $Diff(a,b)$ is defined as: if a=b, then $Diff(a,b)=0$; otherwise $Diff(a,b)=1$. The numeric area of plain disagreement measure is [0, 1]. If two classifiers give the same output for each sample, the measure is 0; when the forecast value for each sample is not always the same, the measure is 1. The greater the measure is, the greater the diversity between classifiers is.

2.2 Diversity measure

Kappa measure on disagreement of two classifiers i and j is shown as formula (2). For $x\in X$, N_{ab} is defined as: the number of samples when the output of classifier i is $h_i(x)=a$, and the output of classifier j is $h_j(x)=b$. N_{a*} is the number of samples recognized by classifier i as class a, N_{*a} is the number of samples recognized by classifier j as class a, N is the total number of samples, and m is the number of classes.

$$\Theta_1=\frac{\sum_{i=1}^{m}N_{ii}}{N};$$

$$\Theta_2=\sum_{a=1}^{m}(\sum_{b=1}^{m}\frac{N_{ab}}{N}\cdot\sum_{b=1}^{m}\frac{N_{ba}}{N})=\sum_{a=1}^{m}(\frac{N_{a*}}{N}\cdot\frac{N_{*a}}{N})$$

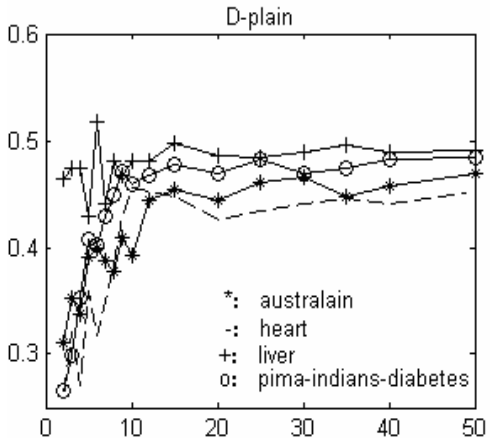
$$D\_kappa(i,j)=\frac{\Theta_1-\Theta_2}{1-\Theta_2} \qquad (2)$$

where Θ_1 is the probability estimates of two classifiers on agreement, Θ_2 is the probability estimates of accidental agreement, which is the revise of Θ_1. When $\Theta_1=\Theta_2$, $D\_kappa(i,j)=0$, and two classifiers are considered as disagreement, that is, they are independent of each other; when Θ_1=1, $D\_kappa(i,j)=1$, and the classification output of the two classifiers are considered as agreement for the all input model. The value of $D\_kappa(i,j)$ can be minus, but it is few minus. The diversity between classifiers is in inverse proportion to this measure, and kappa diversity measure can trace negative correlation.

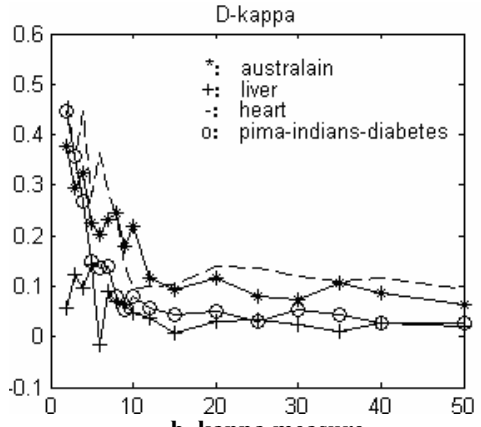
The studies given above are all the diversity measure functions between paired classifiers. The average of diversity measure values of all classifier pairs is calculated as the diversity measure in ensemble in applications. Use D_kappa as an example, the method by which the average of diversity measure values of classifier pairs is calculated is:

$$\overline{D\_kappa}=\frac{2}{L(L-1)}\sum_{i=1}^{L-1}\sum_{j=i+1}^{L}D\_kappa(i,j) \qquad (3)$$

Four test sets of UCI data warehouse are used to test the diversity of ensemble classifiers. They are Australain, Heart, liver-cost-bupa and pima-indians-diabetes. AdaBoost algorithm is used to train 50 ensemble classifiers for each data set in the experiment, and BP-based neural network is used as the base classifiers. Measure function on disagreement between classifiers mentioned above is used to figure out the tendency chart which shows the values of diversity measure changes with the change of the number of classifiers. The x axis figures the number of classifiers and the y axis figures the diversity measure of classifiers in figure 1.



a. plain disagreement measure



b. kappa measure

Figure 1. The diversity measure trend with the number of classifiers on four test sets

From a and b figure of the figure 1, we can see that multiple classifiers diversity produced by AdaBoost algorithm is not increased all through with the increasing of the classifier numbers. At the beginning when the new classifiers join in, the ensemble diversity is increased rapidly. But when the number of classifiers reaches a certain amount, the diversity becomes stable gradually and will not be increased. The conclusion above is supported by the result of several test sets. This is because it will iteratively produce more similar classifiers by repeating sampling from similar distribution data sets and make the effect of some classifiers having diversity weaker. Therefore, we need to prune the ensemble classifiers on the basis of not reducing the performance of ensemble classification, and save the classifiers having diversity.

3. Subset partition method based on coincident errors measure of classifiers

The enhancement of recognition performance in ensembles learning depends more on the characteristic [2] that each classifier makes a wrong decision-making simultaneously for the same or similar samples compared to the fusion method, that is, if the same wrong decision-making samples made by each classifier simultaneously are less, the recognition performance of multiple classifiers will be enhanced more; contrarily, if the probability for each classifier making the same wrong decision-making simultaneously is larger, it's harder to enhance the recognition performance of ensembles learning. For two classifiers, classification diversity for the same sample shows the degree of their association. So we divide the classifier set into several classifier subsets according to the diversity measure between classifiers. Classifiers in the same subset have a lower diversity and classifiers in different subsets have a greater diversity. We can study the choice for the ensemble classifiers according this.

Suppose that E is the collection of L ensembles learning and training classifiers. $E = \{h_1, h_2, ..., h_L\}$ .The purpose of pruning is to seek a classifier subset E^* having optimal performance ($E^* \subseteq E$). E^* can be obtained by evaluating the classification accuracy of all subsets of E .But the possible result may be $\sum_{i=1}^{L} \binom{L}{i}$. When the number N of examples is larger, there are many such enumerating subsets. So we hope to find a method which can find the classifier set of optimal combination without the need of enumerating all subsets of E .

E is divided into M disjoint classifier sets. $E = \{E_1, E_2, ..., E_M\}, E_i \subseteq E$, and $E_i \bigcap E_j = \phi$, $\forall h_r, h_m \in E_i, h_n \in E_j$, we have

$$P(h_r \ error, h_m \ error) > P(h_r \ error, h_n \ error) \qquad (4)$$

Where $P(h_r \ error, h_m \ error)$ expresses the compound errors probability of forecast classifiers h_r , h_m . The formula above shows that the compound errors probability of any two classifiers which belong to the same classifier subset is greater than the multi-probability of any two classifiers which belong to two different classifier sets, and those classifiers which have greater simultaneous errors probability for the same or similar samples will be included in the same subset. Therefore, we need to solve the following key issues: How to satisfy each classifier partition E_i in formula (4), and how to choose the classifiers from each classifier partition to constitute the optimal classifier set.

For the input training set X, L classifier sets $h_t, t = 1,2,...,L$ is produced by the method of AdaBoost.M1. The purpose of pruned classifier is: Minimize the number of classifiers under the premise of not affecting classification accuracy, that is, recognize a minimum classifier set which can achieve the forecast accuracy from $h_t, t = 1,2,...,L$. In order to achieve the forecast ensemble classification accuracy, the accuracy and diversity of classifiers must be kept.

Suppose that $H = \{h_t, t = 1,2,...,L\}$ is the set of all classifiers. $h_i(x_j)$ expresses the forecast for the sample $x_j \in X, j = 1,...,n$ by classifier h_j , and $y_{i,j} = h_i(x_j)$, $Y_i = \{y_{i,1}, y_{i,2}, ..., y_{i,n}\}, i = 1,2,...,L$ is the forecast by classifier h_i for the training set X. We use clustering algorithm to recognize some classifiers,

which produce similar classification errors for the same or similar input model, and prune redundancy classifiers from the clustering.

In order to divide classifier set into a number of subsets which include similar classifiers, Christino Tamon, etc [9] use standard K-means algorithm. L classifiers produce L forecast vectors $Y_i, i = 1,...,L$. Each Y_i acts as a data model which has $n * m$ properties. Clustering method is used for the data sets which have m properties and L input models, so it has a larger calculating amount. For this reason, an ensemble pruning algorithm based on diversity measure of multiple classifiers is proposed. We use distance function measure in compound error probability clustering between classifiers d_{st}. For $\forall h_s, h_t \in E$:

$$d_{st} = d(h_s, h_t) = 1 - P(h_s \text{ error}, h_t \text{ error}) \quad (5)$$

Formula (5) shows that the lower the coincident errors probability produced by two classifiers is, the greater the diversity between them is. So the classifiers which have greater coincident errors can be included in one class by formula above and the classifiers which produce different errors be included in another class.

The calculating method of compound errors measure $P\_dis(h_i, h_j)$ is proposed. It is the ratio of the number of wrong forecasting samples produced by two classifiers to the total number of observed samples as shown in the formula (6).

$$P\_dis(h_i, h_j) = \frac{N^{00}}{N^{11} + N^{10} + N^{01} + N^{00}} \quad (6)$$

where N^{ab} is the number of samples in test set, a is the classification result produced by classifier i, and b is the classification result produced by classifier j. Denominator is the number of total samples, that is, $N = N^{11} + N^{10} + N^{01} + N^{00}$. The value of a and b can be 0 or 1. a=1 expresses that classifier i classifies correctly, and a=0 expresses it not correctly classifies; b=1 expresses that classifier j classifies correctly, and b=0 expresses it not correctly classifies. The numeric area of coincident errors measure is [0, 1]. If classifiers produce not the same error output for each sample, the measure is 0, and if classifiers produce the same error output for each sample, the measure is 1. $P\_dis(h_i, h_j)$ is used in clustering algorithm as the calculating method of compound error probability in the formula (5). This distance measure can be used for classifier clustering. So classification diversity can be lower in each clustering fascicle and be greater between different clustering fascicles.

4. Pruning algorithm with voting weights reallocation

The purpose of classifiers ensemble is to save the classifiers which have higher classification accuracy and greater diversity in each fascicle, and prune the classifiers which have lower diversity compared to those saved classifiers. So ensemble algorithm with voting weights reallocation is proposed. First the classifiers in each partition fascicle are sorted by their classification accuracy; in each fascicle, it starts from the classifier which has the lowest accuracy. If the diversity between it and other classifiers is larger than a certain pre-set threshold and it satisfies the requirement of the classification accuracy, then the classifier will be saved, otherwise deleted. That is, the classifier fascicles $Cl_m, m = 1,2,...,k$ are ascending sorted by the classification accuracy. The classifier having the highest accuracy is $h_{\max}$. $\forall h_t \in Cl_m, h_t \neq h_{\max}$, the difference between h_t and the recognition rate of $h_{\max}$ is calculated. If the difference is larger than the pre-set threshold, the classifier will be pruned.

After the classifier has been pruned, voting weights of the other classifiers will be changed. In order to predigest the classifier pruning in each clustering, here it is realized by assigning each voting weight to each classifier. Suppose pruned classifier set is $H = \{h_t \mid t = 1,2,...,L\}$, the voting weight of classifier h_t is β_t. The voting weight of pruned classifier shift to other classifiers in H. Its weight can be partitioned according to the diversity measure of the pruned classifiers and the classifiers having not been pruned in partition, and shift the partitioned weight to related classifiers.

The correlation coefficient between pruned classifiers and the classifiers having not been pruned is used here as the diversity measure between two classifiers. The correlation coefficient [7]between classifier i and j $D\_\rho(i, j)$ can be measured by formula (7).

$$D\_\rho(i, j) = \frac{N^{11}N^{00} - N^{01}N^{10}}{\sqrt{(N^{11} + N^{10})(N^{01} + N^{00})(N^{11} + N^{01})(N^{10} + N^{00})}} \quad (7)$$

where N^{ab} is defined the same as the one in formula (6). When a classifier is pruned, the classifiers which are similar to the pruned classifier will gain a greater weight. It makes the process of weight assigning contribute to producing more reliable classifiers ensemble.

5. Experiment analysis

The ensemble learning base classifier uses Neural Networks of forward feedback and BP learning algorithm. Two test sets of UCI data set which are iris and glass and Radar Radiant Point data are used in the experiment. In Radar Radiant Point data, the seven types of radar detection data of system 6 (general pulse) are chosen. The information of these three test sets is shown in table 1.

Table 1 Three test sets

Data Set	# Instances	#Class	#Attribute	#test Instances
Iris	150	3	4	30
Glass	214	6	9	42
Radar	840	7	3	168

Table 2 Pruning level and ensemble classification errors

Data Set	Glass L=40	Iris L=40	Radar L=40
Accuracy of entire ensemble	0.738	0.967	0.994
Pruning Level	50%	65%	62.5%
Accuracy of pruned ensemble	0.723	0.967	0.993
Number of clusters	4	3	3

In table 2, L expresses the number of the classifiers. The result shows that when the pruning rate is between 50%~60%, the ensemble classification accuracy is basically commensurate to the ensemble recognition rate of all classifiers.

6. Summary

A diversity measure describing function based on the diversity between classifiers in ensembles learning is proposed in this paper, and a pruning method for ensemble classifiers based on diversity gene is proposed also. For the pruned classifiers, the method of voting weights reallocation is used to gain the optimal classifier subsets and related weight assigning. The UCI data depository and Radar Radiant Point data are used as test data, and the result of experiment show that classifiers ensemble with pruning can get similar classification accuracy as accuracy of entire classifier integration and reduce classification time.

References

[1] Hansen, L. Salamon, P. Neural network ensembles, IEEE Transactions on Pattern Analysis and Machine Intelligence, 12, 993-1001. 1990.

[2] Tumer, K., Ghosh, J. Error correlation and error reduction in ensemble classifiers, Connection Science, Special Issue on Combining Artificial, Neural Networks: Ensemble Approaches, 8(3-4),385-404. 1996.

[3]Ludmila I. Kuncheva, A Theoretical Study on Six Classifier Fusion Strategies, IEEE Transactions on Pattern Analysis and Machine Intelligence, Nageswara SV Rao, v.23 n.8, p.904-909, August 2001.

[4] Dragos D. Margineantu, Thomas G. Dietterich: Pruning Adaptive Boosting. Proceedings of the Fourteenth International Conference on Machine Learning (ICML 1997), Nashville, Tennessee, USA, 211-218. 55, 1997.

[5] Lazarevic, A.,Obradovic, Z. The Effective Pruning of Neural Network Ensembles, in Proceedings of the IEEE International Joint Conference on Neural Networks, Washington D.C.,796-801,2001.

[6]Fang min. Study of integration method for multiple classifiers on ensemble learning. Systems Engineering and Electronics.Vol.28 No.11, p 1759-1761+1769,2006.

[7] G. Zenobi, P. Cunningham. Using diversity in preparing ensembles of classifiers based on different feature subsets to minimize generalization error, in: L.D. Raedt, P.A. Flach (eds.), Proc. ECML 2001 12th European Conf. On Machine Learning, LNCS 2167, Springer, 2001, pp. 576-587.

[8]T.G. Dietterich. An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization, Machine Learning 2000,40(2):139-157.

[9]Christino Tamon，Jie Xiang , On the Boosting Pruning problem, Proceedings of 11th European Conference on Machine Learning, Springer, 404-412. 2000.