Combining Multiple Classifiers: Diversify with Boosting and Combining by Stacking

Nima Hatami [†] **and Reza Ebrahimpour** [††],

[†] Department of Electrical Engineering, Shahed University, Tehran, Iran
[††] School of Cognitive Sciences, Institute for Studies on Theoretical Physics and Mathematics, Tehran, Iran and
Department of Electrical Engineering, Shahid Rajaee University, Tehran, Iran

Summary
Combining multiple classifiers is one of the most important topics in pattern recognition. In this paper the idea is to combine classifiers with different error types by a learnable combiner which is aware of the classifiers' expertise, so that the variance of estimation errors is reduced and the overall classification accuracy is improved. To achieve diverse base classifiers we use the boosting method in which the classifiers are trained with differently distributed training sets. And to combine the diverse base classifiers, considering their area of expertise, we use stacked generalization method which minimizes the generalization error by a classifier at a second layer to learn the type of errors made by the first layer classifiers. The proposed model is experimented with the SATIMAGE data from ELENA database. Experimental results show that the proposed model outperforms the stack and boosting methods with higher classification accuracy.

Key words:
Combining classifiers, stacked generalization, Boosting.

1. Introduction

Combining classifiers to achieve higher accuracy is an important research topic with different names such as combination of multiple classifiers, committee machine, classifier ensembles and classifier fusion [1-13]. They are also, proposed to improve the classification performance of a single classifier [14-16]. In classification combinations, it is expected that the differently trained classifiers converge to different local minima on the error surface, and the overall performance is improved by combining the outputs in some ways. A major factor behind any improvement is the generalization performance, which addresses the problem of how to develop a classifier with a finite number of training samples to achieve optimal performance on samples that are not included in a training set. Another factor is the diversity in the classifier opinions [17], for which, methods such as boosting and bagging [18] have been introduced.

Stacked generalization proposed by Wolpert [19], is a layered architecture. The classifiers at the Level-0 receive the original data as their input, and each classifier outputs a prediction for its own sub problem. Successive layers receive the predictions of the layer immediately preceding it as an input and finally a single classifier at the top level outputs the final prediction. Stacked generalization attempts to minimize the generalization error by using classifiers at higher layers to learn the type of errors made by the classifiers immediately below.

In another approach, boosting [20, 21] is one of the most important developments in classification methodology. The intuitive idea behind Boosting algorithm is to train a series of diverse classifiers and to iteratively focus on the hard to learn training examples. It is implemented by two steps, training a number of classifiers with the various versions of the training sample and then combining these classifiers to produce a more powerful one. In the first step, the boosting method generates different training sets using the performance of former classifiers so that training instance that are wrongly predicted by former classifiers will play more important roles in the training of later classifier. It is the special technique that makes boosting effective for the hard-to-learn sample and obtains remarkable improvement in performance.

We applied proposed model to the SATIMAGE data from ELENA database [22]. The experimental results show that proposed model indeed improve the classification accuracy compared with the original stacking and boosting methods.

This paper is organized as follows. In Section 2, we provide the theory of Boosting method. Section 3 and 4 explain the stacked generalization and proposed methods respectively. Experimental results are given in Section 5 to show the effectiveness of the proposed method. Conclusions are given in Section 6.

Manuscript received January 5, 2007
Manuscript revised January 25, 2007

2. Boosting method

The boosting method introduced to improve the performance of weak classifiers [23].Its theoretic basis relies on a proof of the equivalence of the strong and weak PAC learning models. In the standard PAC (Probably Approximately Correct) model, for any distribution of pattern and for arbitrary small δ and ε, the learner must be able to produce a hypothesis about the underlying concept, with an error rate of at must ε with approximately of at least (1-δ). The weak PAC model, however, just requires: ε<1/2, i.e. slightly better than a random guess on this two-class model.

Schapire proved the equivalence of the two models by proposing a technique for converting any weak learning algorithm (on any distribution) to a strong learning algorithm. He termed this provably correct technique-boosting. The basis of the technique is creating different distributions on which different sub-hypotheses are trained. Schapire has proved that if three such weak sub-hypotheses, which have an error rate of $\alpha < 1/2$ (on the respective distribution) are combined, then the resulting ensemble hypothesis will have an error rate of $3\alpha^2-2\alpha^3$ which is smaller than α. schapire suggested hierarchical combinations of classifiers, such that an arbitrarily low error rate can be achieved.

A procedure for creating appropriate distributions is the following: As shown in Figure 1, in conventional boosting, the algorithm relies on continuously changing the distribution of training set so that those that are frequently misclassified get important role: this way, new classifiers that are added to the ensemble are more likely to classify those hard examples correctly. In the end, boosting algorithm predicts one of the classes based on the sign of a linear combination of the weak classifiers trained at each step. Thus, multiple copies of the "difficult" training samples are likely to appear in the next training set, focusing the "expertise" of the classifier onto a problematic region in the feature space.

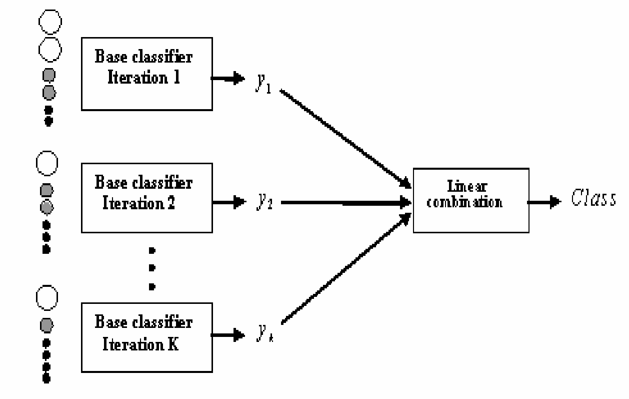

Base classifier Iteration 1
y_1
Base classifier Iteration 2
y_2
Base classifier Iteration K
y_k
Linear combination
Class

Fig. 1. Block diagram of a multiple classifiers systems based on Boosting

3. Stacked generalization method

The method of stacked generalization provides a way of combining trained networks together which uses partitioning of the data set (in a similar way to cross-validation) to find an overall system with usually improved generalization performance.

Consider the modular network system shown in figure 2. Here we see a set of K "level-0" network N_1^0 to N_k^0 whose outputs are combined using a "level-1" network N^1.

The idea is to train the level-0 networks first and then examine their behavior when generalizing. This provides a new training set which is used to train the level-1 network.

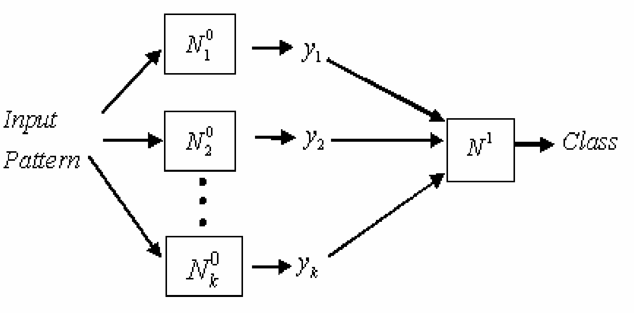


Fig. 2. Block diagram of a multiple classifiers systems based on stacked generalization

4. Proposed model

Our proposed model is based on the stacked generalization method in which, the boosting technique is used to train the base classifiers. This way, the base classifiers achieve more diversity and each of them covers different area in error surface. Therefore in the combining stage, the error surface is covered in such a way that the classification accuracy is efficiently improved.

Wolpert suggests that the level-0 networks should contain a wide variety of diversity in classifiers, while (so that) the level-1 network should provide a relatively smooth function and hence should have a relatively simple structure. In our proposed model, we provide more diverse classifiers at the level-0 by apply the boosting algorithm instead of the cross-validation of the conventional style. We will show that, proposed model have a better performance than original stack because of applying boosting as a robust method for diversify level-0 classifiers.

As shown in figure 3, in level-0 of our model, we use a version of the Boosting algorithms, which classifiers are trained serially. After training the first classifier, a copy of the "difficult" training samples is added to the next training set which is used to train the second classifier. This procedure is repeated for all base classifiers. After the training the level-0 networks, they are run with the training set to provide a new training set for the level-1 network. This generates a single pattern for a new data set which will be used to train the level-1 network. The inputs of this pattern consist of the outputs of all the level-0 networks, and the target value is the corresponding target value from the original full data set.

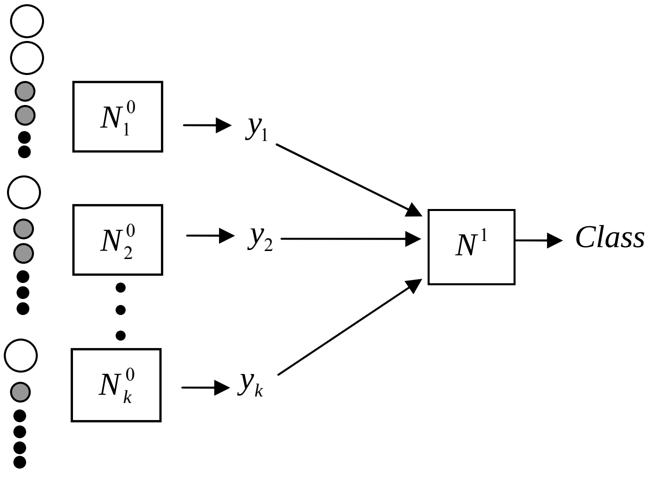


Fig. 3. Sketch of proposed model

5. Experimental Results

Our proposed model was experimented with the SATIMAGE data from ELENA database. The Satimage data was generated from Landsat Multi-Spectral Scanner image data. It consists of 6435 pixels with 36 attributes (4 spectral bands × 9 pixels in a 3 × 3 neighborhood). The pixels are crisply classified in 6 classes, and are presented in random order in the database. The classes are: red soil (23.82%), cotton crop (10.92%), grey soil (21.10%), damp grey soil (9.73%), soil with vegetation stubble (10.99%), and very damp grey soil (23.43%). What makes this database attractive is: large sample size; numerical, equally ranged features; no missing values; and compact classes of approximately equal size, shape and prior probabilities. In our experiments we used features #36 and we display only the 91.58% correct on the test sets, which have not been seen during training of either the base classifiers or the level-1 fusion model.

To evaluate the performance of the proposed model, we compared it with the stack and boosting methods. The proposed model has 3 base classifiers, as the laval-0 network. The network's input layers had 36 nodes and their single hidden layers consisted of 19, 20 and 21 nodes respectively and 6 output nodes, corresponding to the 6 classes of the dataset. The level-1 network, as the combiner, had 18 input nodes, corresponding to the outputs of the tree level-0 networks and 16 hidden neurons. The 6-dimentional output vector was used to extract the total output classes and its confidence level.

For the two other models, the boosting and stacked generalization, the similar network structure was used. In boosting, three classifiers were trained on different distribution and their outputs were combined by the averaging rule and for the stack method, the tree base classifiers were trained in its conventional style, that is, the cross-validation technique.

The optimum number of hidden neurons was found experimentally by training and testing the networks with different structures. The number of neurons in the back-propagation network hidden layer is determined using a cascade learning process [24]. The cascade learning process is constructive, starting with an empty hidden layer and adding neurons to this layer one at a time. The addition of hidden neurons continues until there is no further improvement in network performance and therefore the optimum number of hidden neuron founded.

As shown in Figure 4, classifiers in Fig. 4B which is generated with the boosting method are more diverse than classifiers in Fig. 4A which generated whit cross-validation used in stack method. In the other word, the classifiers expertise in Fig.4B is more different in each class as compared with same class in other classifier. Therefore the boosting method is able to create more different classifiers with respect to the stack method. Since in combining classifiers, having the more diverse and independent classifiers is desirable to achieve better performance, we use boosting method instead of cross-validation in proposed method, for having an overall classifier with higher accuracy.

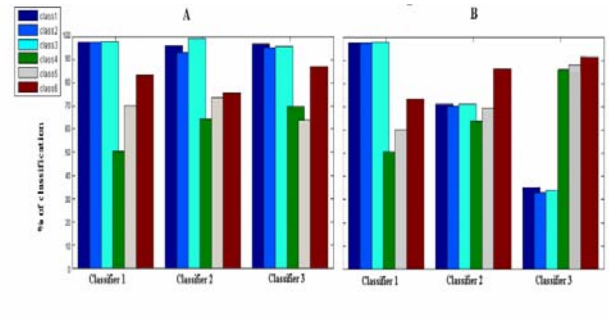


Fig. 4. Bars denote the classification accuracy of base classifiers for each classes. A) Base classifiers generated with cross-validation. B) Base classifiers generated with boosting method.

We apply proposed model, boosting and stack method, to the Satimage dataset and compare their results to find out the robustness method. All classifiers used were feed-forward neural network, trained via back-propagation algorithm. Since the neural models are stochastic methods with randomly determined starting points, the results are averages of ten repetitions on data set. Table 1 presents the performance of these three ensemble methods. The results indicate that the performance of proposed model is significantly better than boosting and stack methods.

Table 1: Performance of various ensembles on Satimage dataset

	Boosting	Stack method	proposed model
Classifier 1	87.83 %	84.86 %	87.83 %
Classifier 2	88.57 %	86.66 %	88.57 %
Classifier 3	88.11 %	86.48 %	88.11 %
Ensemble	89.37 %	88.41 %	**91.58** %

6. Conclusion

In this paper, we proposed new model which used boosting as a diversifying approach in the base classifiers, instead of cross-validation for more diverse and independent base classifiers. Unlike the conventional linear combiner for instance averaging rule, used in Boosting, we employed Stacking method. Our experimental results on Satimage dataset confirm that the base classifiers, generated with boosting are more diverse than that generated with cross-validation approach. Our results also show the robustness of our model in comparison with original boosting and stacking methods in classification accuracy.

References

L. Lam and C.Y. Suen. Optimal combination of pattern classifiers. Pattern Recognition Letters, 16:945-954, 1995.

[2] G. Rogova. Combining the results of several neural network classifiers. Neural Networks, 7:777-781,1994.

[3] K. Woods, W.P. Kegelmeyer, and K. Bowyer. Combination of multiple classifiers using local accuracy estimates. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19:405-410, 1997.

[4] L. Xu, A. Krzyzak, and C.Y. Suen. Methods of combining multiple classifiers and their application to handwriting recognition. IEEE Transactions on Systems, Man, and Cybernetics, 22:418-435, 1992.

[5] J. Kittler, M. Hatef, R.P.W. Duin, and J. Matas. On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(3):226-239, 1998.

[6] S.-B. Cho and J.H. Kim. Combining multiple neural networks by fuzzy integral and robust classification.IEEE Transactions on Systems, Man, and Cybernetics, 25:380-384, 1995.

[7] P.D. Gader, M.A. Mohamed, and J.M. Keller. Fusion of handwritten word classifiers. Pattern Recognition Letters, 17:577-584, 1996.

[8] M. Grabisch and F. Dispot. A comparison of some for fuzzy classification on real data. In 2nd International Conference on Fuzzy Logic and Neural Networks, pages 659-662, Iizuka, Japan, 1992.

[9] J.M. Keller, P. Gader, H. Tahani, J.-H. Chiang, and M. Mohamed. Advances in fuzzy integration for pattern recognition. Fuzzy Sets and Systems, 65:273-283, 1994.

[10] I. Bloch. Information combination operators for data fusion: a comparative review with classification. IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, 26:52-67, 1996.

[11] H. Drucker, C. Cortes, L.D. Jackel, Y. LeCun, and V. Vapnik. Boosting and other ensemble methods. Neural Computation, 6:1289-1301, 1994.

[12] E. Filippi, M. Costa, and E. Pasero. Multy-layer perceptron ensembles for increased performance and fault-tolerance in pattern recognition tasks. In IEEE International Conference on Neural Networks, pages 2901-2906, Orlando, Florida, 1994.

[13] S. Haykin, Neural Networks, Prentice-Hall, New Jersey, 1999.

[14] T. K. Ho, J. J. Hull, and S. N. Srihari, Decision combination in multiple classifier systems, IEEE Trans. Pattern Anal. Machine Intell., vol. 10, pp. 66–75, 1994.

[15] E. M. Kleinberg, Stochastic discrimination, Ann. Math. Artificial Intell., vol. 1, pp. 207–239, 1990.

[16] L. Xu, A. Krzyzak, and C. Y. Suen, Methods of combining multiple classifiers and their applications to handwriting recognition, IEEE Trans. Syst., Man, Cybern., vol. 22, pp. 418–435, 1992.

[17] K. Ali, M. Pazzani, On the link between error correlation and error reduction in decision trees ensembles, Technical Report 95-38, Department of Information and Computer Science, University of California, Irvine, 1995.

[18] L. Breiman, Bagging predictors, Mach. Learning, 24:123–140, 1996.

[19] D. Wolpert, Staked generalization, Neural Networks, vol. 5, no. 2, pp. 241–259, 1992.

[20] Y. Freund, R. Schapire, Experiments with a new boosting algorithm, in: L. Saitta (Ed.), Proceeding of the Thirteenth International Conference over Machine Learning, Morgan Kaufmann, San Francisco, 1996, pp. 148–156, 1996.

[21] Y. Freund, R. Schapire, A decision theoretic generalization of on-line learning and an application to boosting, J. Comput. Syst. Sci. 55 (1997) 119– 139.

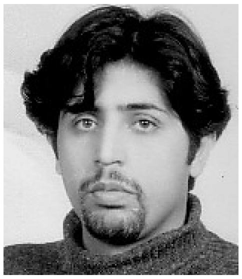
[22] ELENA Project, http://www.dice.ucl.ac.be/neuralnets/ Research/Projects/ELENA/elena.htm, 2003.

[23]R. Schapire, The strength of weak learnability. Machine Learning, 5(2):197-227, 1990

[24] Neural Computing. Neural Ware, Pittsburgh, PA, 1991

Nima Hatami received the B.S. degrees in Electrical Engineering from Shahid Rajaee University, Tehran, Iran in 2006. He is currently working toward the M.S. degree in the Department of Electrical Engineering, University of Shahed, Tehran, Iran.

His research interests include Machine learning, Neural networks and Multiple Classifier Systems.



Reza Ebrahimpour received the B.Sc. degree in electrical engineering from Mazandaran University (1999), and the M.S. degrees in Biomedical Engineering from Tarbiat Modares University, Tehran, Iran in 2002. He is currently working toward the Ph.D. degree in the Computational neuroscience, School of Cognitive Sciences, Institute for Studies on Theoretical Physics and Mathematics, Tehran, Iran.

In 2003, he joined the Department of Electrical and Computer Engineering at the University of Shahid Rajaee, Tehran, Iran.

His research interests include Machine learning, Neural networks, Multiple Classifier Systems, Artificial intelligence and Human and biological vision.