Expert Systems with Applications 38 (2011) 8414–8418

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Extended decision template presentation for combining classifiers

Mehdi Salkhordeh Haghighi *, Abedin Vahedian [1], Hadi Sadoghi Yazdi [1]

Department of Computer Engineering, Ferdowsi University of Mashhad, Mashhad, Iran

ARTICLE INFO

Keywords:
Decision template
Classifiers fusion
Classifiers combiner

ABSTRACT

In this paper, a new method in classifier fusion is introduced for decision making based on internal structure of base classifiers. Amongst methods used in combining classifiers, there are some methods which work on decision template as a tool for modeling behavior of base classifiers in order to label data. This tool models their behavior only based on their final outputs. Our new method, introduces a special structure for decision template such that internal behavior of a neural network base classifier can be modeled in a proper manner suitable for classifiers fusion. The new method builds decision template for each layer of the neural network including all hidden layers. Therefore, the process of making decision in each base classifier is also available for classifiers fusion. Efficiency of the new method is compared with some known benchmark datasets to show how it can improve efficiency of classifiers fusion.

1. Introduction

Classification is one of the most frequently encountered decision making processes in a wide range of applications. A classification problem occurs when an object needs to be assigned into a predefined group or class based on various properties or features. Many problems in business, science, industry, military, security and medicine can be treated as classification problems.

Many classification techniques have been used in various applications over the last decade. Moreover, many optimization techniques have also been introduced to enhance efficiency and accuracy of classification. Among these, ensemble techniques or MCSs[2] also named classifiers fusion or combining classifiers have special interest. As a general rule, the combinational methods have more power, robustness, resistance, accuracy and generality rather than single classification (Analoui, 2008). The motivation for this procedure is based on the intuitive idea that by combining the outputs of several individual predictors one might improve performance of a single generic one (Krogh & Vedelsdy, 1995).

However, the idea of performance improvement in MCS has been proved to be true only when the combined base classifiers are accurate and diverse enough, which requires an adequate tradeoff between these two conflicting conditions (Navone, Granitto, Verdes, & Ceccatto, 2001). Also, Kuncheva (2005) using Condorcet Jury theorem (Shapley & Grofman, 1984) has shown that combination of classifiers can usually operate better than a single classifier. It means that if classifiers with more diversity are used in the ensemble, then total error can considerably be reduced. Drucker, Schapire, and Simard (1993) attempted to gain a good compromise between these properties including elaborations of bagging (Breiman, 1996), boosting (Freund & Schapire, 1995) and stacking (Wolpert, 1992) techniques.

In general, three types of strategies in combining classifiers can be identified (Xu, Krzyzak, & Suen, 1992). In the first type each classifier produces output as a single class label such that these labels have to be combined to make final decision (Battiti & Colla, 1994). In the second type outputs of the classifiers are sets of class labels ranked in the order of likelihood (Tumer & Ghosh, 1995) and the third type involves the combination of real valued outputs for each class by the respective classifiers, most often posterior probabilities (Jacobs, 1995), sometimes evidences (Rogova, 1994).

Nevertheless, there are generally two types of combination named classifier selection and classifier fusion (Woods, Kegelmeyer, & Bowyer, 1997). The presumption in classifier selection is that each classifier is an expert in some local area of the feature space. When a feature vector is submitted to each classifier, the classifier designated for the vicinity of x is given the highest credit in assigning the class label to x. Therefore, exactly one classifier is responsible to make the final decision, as in Ng and Abramson (1992), or more than one base classifier, as in Jacobs, Jordan, Nowlan, and Hinton (1991), Alpaydin and Jordan (1996). Classifier fusion assumes that all classifiers are trained over the whole input feature space, and are thereby considered as competitive rather than complementary (Rastrigin & Erenstein, 1982; Xu et al., 1992).

However, between all the methods introduced for combining classifiers, DT based methods have special features suitable for wide range of applications. DT method models behavior of base classifiers for making decision on input data. This modeling

* Corresponding author. Tel.: +98 511 9151135801; fax: +98 511 8763306.
E-mail addresses: haghighi@ieee.org (M.S. Haghighi), vahedian@um.ac.ir (A. Vahedian), sadoghi@sttu.ac.ir (H.S. Yazdi).

[1] Tel./fax: +98 511 8763306.

[2] Multiple Classifier System.

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.01.036

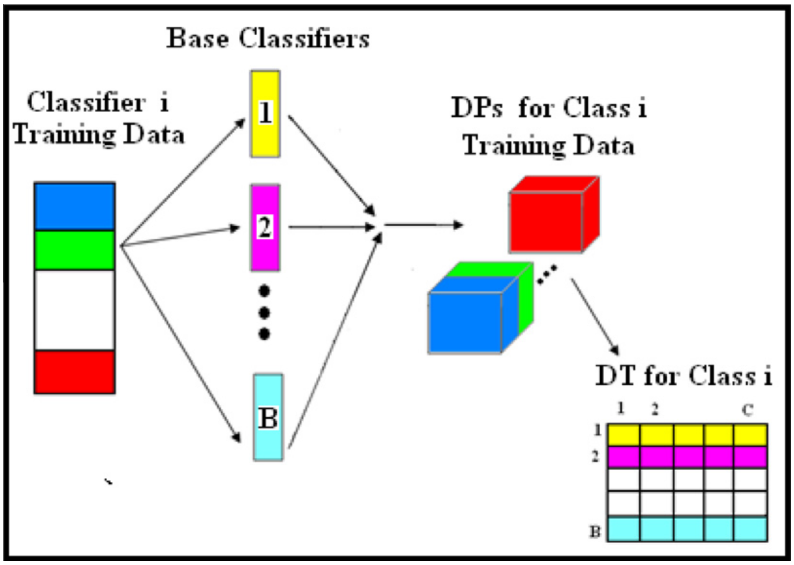

Fig. 1. Construction of DT array for Class I data.

scheme is constructed based on final outputs of base classifiers. Operation of the method is based on a set of DT matrices. DTs make a robust classifier fusion scheme that combines classifier outputs by comparing them with a characteristic template for each class. DT based fusion uses all classifier outputs to calculate the final support for each class, which is in sharp contrast to most other fusion methods which use only the support for that particular class to make their decision.

DT based methods have been using in wide range of applications for many years. In our research, DT structure is used for modeling behavior of base classifiers in a MCS not only based on their output results but also based on internal processing and partial paths taken for making final decision. Using this method to deeply describe partial decisions made inside the structure of base classifiers can make classifier combiner operation more robust and efficient. The motivation for these properties lies in the fact that the classifiers combiner has more partially processed data with more options to make decisions.

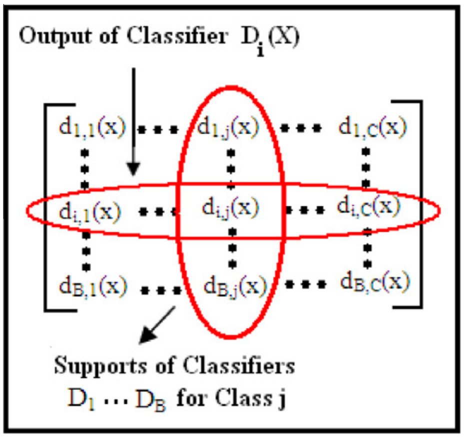

Fig. 2. Basic DP structure.

2. Problem definition

As mentioned, the primary goal of a MCS is to improve overall efficiency of base classifiers in making decisions over input data. One of the methods widely used in MCSs is DT method. Basic DT method constructs decision template array for each class of training data only based on final outputs of the base classifiers. Therefore, the combiner uses these arrays to make final decisions for test data. Basically, for the combiner to have more primary tools for making decisions, a new structure for constructing DT is proposed. The DT is constructed not only based on final outputs of base classifiers, but also based on internal decision paths each base classifier follows to make final decision. In the next section, more details about the new DT are discussed.

2.1. Preliminaries

One of the aspects that deeply affect performance of the MCS is that the behavior of base classifiers is modeled in a proper manner such that in almost all input space, the behavior is completely known. However, since such exact model is hard to find, the base classifiers should be trained such that each of them can model part of the input space more efficiently. Moreover, this type of modeling would be more effective when internal path of decision making in each base classifier is also modeled.

Therefore, the primary goal of this research is to focus on internal behavior of base classifiers for training and testing data such that the combiner part of MCS posses more effective tools to make decisions for labeling input data. This information is used to improve overall generalization capability of the system. For this to happen, neural network based classifiers are used as base classifiers. In addition, different strategies for final combiner are tested to find a suitable test bed for comparison.

2.1.1. Standard DT method

Generally, in standard DT method, DT is constructed for each class of training data. Fig. 1 shows structure and process of making a standard DT. In this case, to build a DT for class I training data, DP is constructed for each data in this class, according to Fig. 2. Next, average value of these DPs is called DT for class I. The same procedure is done for all other classes of training data. The idea behind this DT is to remember the most typical DP for each class. In the test phase, for a test data, DP is constructed, and then compared to each of the DTs based on some similarity measure. Final decision would be the closest match to label the test data.

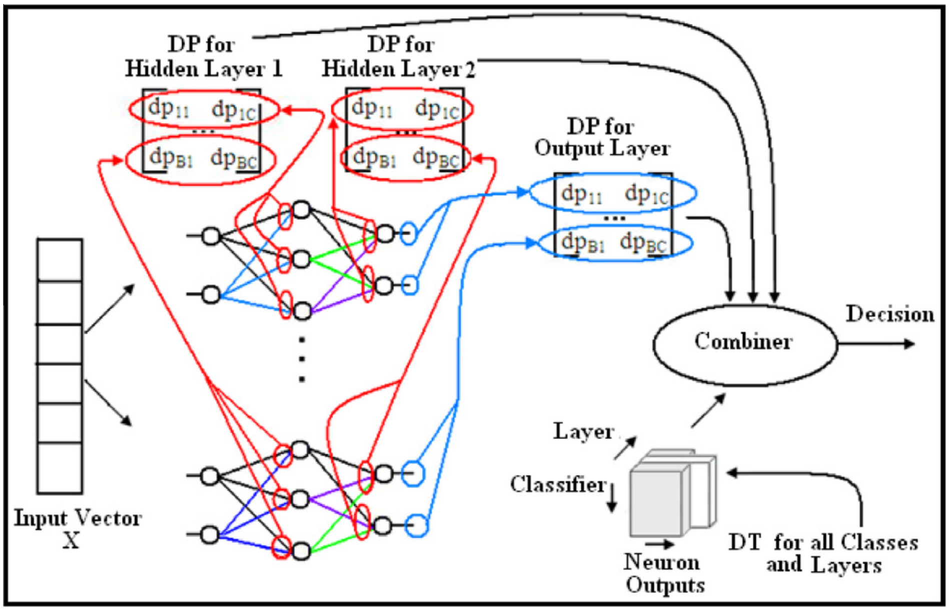

Fig. 3. Structure of the new system.

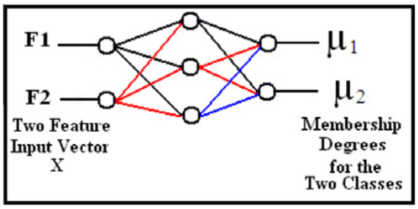

Fig. 4. Structure of a NN based classifier.

The process exhibits some weaknesses. For instance, final decision is made based on some distance measure between DP of input test data and each one of DTs. These distance measures compare distances between the DP and each DT according to two representative points. Using a point as a representative for a group of data does not yield the required accuracy and robustness while it cannot either describe properties of the data in a proper way.

2.1.2. Fusion system structure

Fig. 3 shows structure of our classifiers fusion system from a general view of operation. In this figure, X is a K featured input data, B is the number of base classifiers, and C is the number of classes. The base classifiers used in the system are neural networks, all of which with the same internal structure, as shown in Fig. 4. This means that the number of hidden layers and the number of neurons in each layer for all the base classifiers are the same. In order to keep diversity, operation of the base classifiers should be different in input space by using different training data.

During test step, input vector X is used as input data and DPs for each layer of base classifiers are formed. Since the number of hidden layers and the number of neurons in each layer for the base classifiers are the same, for a hidden layer with N neurons, DP would be a $B \times N$ matrix. However, by using the special structure designed for DP and DT, it would be possible to use base classifiers with different number of neurons in their corresponding hidden layers. It is clear that DTs are formed during base classifiers training. At the end, the combiner makes final decision based on the DTs and DPs formed. Output of the combiner would be a label for the input data. Different strategies for combining these DTs and DPs can be used. In standard DT method, some distance measures are used to determine similarity between each DT and DP.

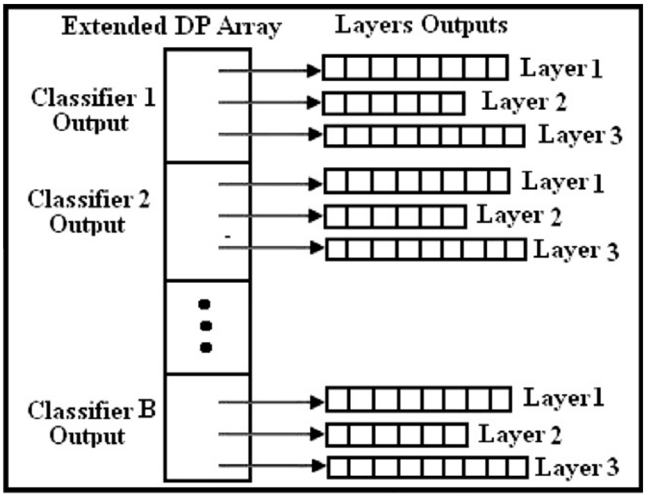

Fig. 5. DP structure used in the new system.

The combining mechanism used in the MCS is based on Euclidean distance measure. To do this, in the test step, after forming DPs for all the layers based on X as input data, the value of DOS[3] for class m in layer k is calculated by Eq. (1). In the equation, $DT^{K,m}$ is the decision template for class m in layer k, N_k is the number of neurons in layer k; DP^k is the decision profile for layer k, C is the total number of classifiers. After computing DOS for each class in each layer, maximum DOS is determined. Finally, voting determines final decision among the maximum values of all layers.

$$DOS_{m,k}(X) = 1 - \frac{\sum_{i=1}^{B}\sum_{j=1}^{N_k}(DT_{i,j}^{k,m} - DP_{i,j}^{k})^2}{B \times N_k} \qquad (1)$$

As a general rule, diversity is a key element in efficiency of MCS. By using a number of different classifiers in a MCS, increase in accuracy and efficiency of the overall system is expected. It is intuitively ac-

[3] Degree of Support.

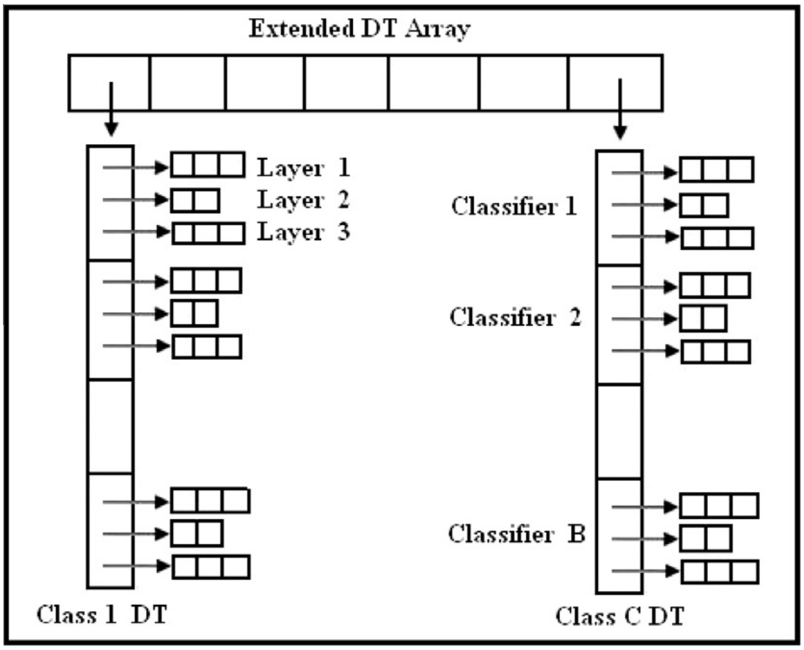

Fig. 6. Structure of the new DT.

Table 1
Properties of some standard datasets.

	# of samples	# of features	# of classes	Address
OCR	1435	64	10	Haghighi@ieee.org
Breast	699	9	2	http://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/
Iris	150	4	3	http://archive.ics.uci.edu/ml/machine-learning-databases/iris/
Glass	214	9	6	http://archive.ics.uci.edu/ml/machine-learning-databases/glass/
Wine	178	13	3	http://archive.ics.uci.edu/ml/machine-learning-databases/wine/

Table 2
Error rate obtained by test data.

	VT	DS	DTED	DTSD	SM	MAX	PT	MIN	The new method
OCR	3.5	3.75	3.75	3.5	3.5	4.75	4.0	5.75	3.15
Breast	3.33	3.33	3.33	3.33	3.33	3.83	3.33	3.83	3.17
Iris	3.57	3.57	2.86	2.86	3.57	3.57	3.57	3.57	2.86
Glass	25	20.67	20.67	26	25	26	35	21.5	19.33
Wine	6.87	6.87	7.5	6.87	6.87	6.87	6.87	6.87	6.87

cepted that classifiers to be combined should be diverse. If they were identical, no improvements would result in combining them. Therefore, diversity among the team has been recognized as a key point. Since the main reason for combining classifiers is to improve their performance, there is clearly no advantage to be gained from an ensemble that is composed of a set of identical classifiers or classifiers that show the same patterns of generalizations. Therefore, different initial values and training sets with different training parameters are used for the base classifiers. As a result, different behavior is expected from these classifiers.

In the new method introduced here, not only a new structure for DT is proposed, but also a new structure for DP is used to match with the new DT structure in the combiner. The criteria for combining is provided in Eq. (2) in which L is the number of layers.

$$\begin{aligned} &DOS_m(X) = max(DOS_{m,k}, k = 1 \ldots L), \\ &Label(X) = argmax(DOS_i(X), i = 1 \ldots C) \end{aligned} \tag{2}$$

Nevertheless, in our new method, DPs are constructed for each of the hidden and output layers of the base classifiers. Therefore, during training, special structure is needed for DPs and DTs such that in test step, the output combiner can easily use them for making decisions. The special DP structure used is shown in Fig. 5.

In this figure, for each classifier, outputs of the neurons of each layer are saved as a vector in corresponding cell of DP array. Since each layer may have different number of neurons, each cell of the DP array is designed such that vectors with different length are stored at the same time. This structure gives the DP more flexibility for using in each MCS with different base classifier structures.

Fig. 6 shows details about the structure of the new DT. In this figure, each cell of DT array stores a DP like structure which is constructed based on the DPs in training step.

3. Implementation and results analysis

Performance of our new system is compared with other methods based on some known benchmark datasets listed in Table 1.

It should be noted that in the system, the structure of all the base classifiers should be the same. Therefore, the number of hidden layers and the number of neurons in each hidden layer should be identical. As a result, comparison of DTs for each layer is done similarly. Eq. (2) indicates labeling process using the DOS obtained by Eq. (1). For each of the data sets listed in Table 1, different groups of data are selected randomly as training and test.

In the first step, base classifiers are trained such that minimum error for each one is produced. Next, using training data, DTs are built for each layer of base classifiers and for each class of training data. These DTs are stored in the data structure shown in Fig. 6.

After the training step, it is time to test performance of MCS with respect to other fusion methods. Performance of the system is measured based on the error rate obtained by test data. As seen in Table 2, comparison has been carried out with these fusion methods: Voting (VT), Dempster Shafer (DS), Decision template and Euclidean distance (DTED), Decision template and symmetric difference (DTSD), simple mean (SM), maximum (MAX), product (PT), and minimum (MIN).

4. Conclusion

In the new method presented, fusion of the decisions made by base classifiers are affected by the internal paths each base classifier follows to produce final decision. Where the base classifiers are neural networks, the intermediate steps each base classifier follows are presented by outputs of hidden layers. In this work, outputs of all hidden layers were preserved in a new structure for DT specially designed for this purpose. Therefore, during test step, these preserved data were used for fusion step such that the combiner could better make final decision to assign a label to the input data. It is expected that the more information available for fusion, the more the decisions are robust. Nevertheless, it is generally expected that efficiency of the combiner increases compared to the methods using only output of the base classifiers for fusion.

References

Alpaydin, E., & Jordan, M. I. (1996). Local linear perceptrons for classification. *IEEE Transactions on Neural Networks, 7*(3), 788–792.

Analoui (2008). CCHR: Combination of classifiers using heuristic retraining. In *International conference on networked computing and advanced information management (NCM 2008)*, Korea, Sep.

Battiti, R., & Colla, A. M. (1994). Democracy in neural nets: Voting schemes for classification. *Neural Networks, 7*(4), 691–707.

Breiman, L. (1996). Bagging predictors. *Machine Learning, 24*(2), 123–140.

Drucker, H., Schapire, R., & Simard, P. (1993). Improving performance in neural networks using a boosting algorithm. In S. J. Hanson, J. D. Cowen, & C. L. Giles, (Eds.), *Advances in neural information processing systems* (Vol. 5, pp. 42–49).

Freund, Y., & Schapire, R. (1995). A decision theoretic generalization of on-line learning and an application to boosting. In *Proceedings of the second European conference on computational learning theory* (pp. 23–37). Springer Verlag.

Jacobs, R. (1995). Method for combining experts' probability assessments. *Neural Computation, 7*(5), 867–888.

Jacobs, R. A., Jordan, M. I., Nowlan, S. J., & Hinton, G. E. (1991). Adaptive mixtures of local experts. *Neural Computation, 3*, 79–87.

Krogh, A., & Vedelsdy, J. (1995). Neural network ensembles cross validation, and active learning. In G. Tesauro, D. Touretzky, & T. Leen (Eds.). *Advances in neural information processing systems* (Vol. 7, pp. 231–238). Cambridge, MA: MIT Press.

Kuncheva, L. I. (2005). *Combining pattern classifiers, methods and algorithms*. New York: Wiley.

Navone, H. D., Granitto, P. M., Verdes, P. F., & Ceccatto, H. A. (2001). A learning algorithm for neural network ensembles. *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial*(12), 70–74.

Ng, K.-C., & Abramson, B. (1992). Consensus diagnosis: A simulation study. *IEEE Transactions on Systems, Man and Cybernetics, 22*, 916–928.

Rastrigin, L. A., & Erenstein, R. H. (1982). Method of Collective Recognition, Energoizdat, Moscow, (in Russian).

Rogova, G. (1994). Combining the results of several neural network classifiers. *Neural Networks, 7*(5), 777–781.

Shapley, L., & Grofman, B. (1984). Optimizing group judgemental accuracy in the presence of interdependencies. *Public Choice, 43*, 329–343.

Tumer, K., & Ghosh, J. (1995). Order statistics combiners for neural classifiers. In *Proceedings of the world congress on neural networks* (pp. I:31–34). Washington DC: INNS Press.

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

Woods, K., Kegelmeyer, W. P., & Bowyer, K. (1997). Combination of multiple classifiers using local accuracy estimates. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 19*, 405–410.

Xu, L., Krzyzak, A., & Suen, C. Y. (1992). Methods of combining multiple classifiers and their applications to handwriting recognition. *IEEE Transactions on Systems, Man and Cybernetics, 22*, 418–435.