International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 4 (2012) pp. 679-686
© MIR Labs, www.mirlabs.net/ijcisim/index.html

Improving Combination Methods of Neural Classifiers Using NCL

Arash Iranzad[1], Saeed Masoudnia[1], Fatemeh Cheraghchi[1], Abbas Nowzari-Dalini[1] and Reza Ebrahimpour[2]

[1]Department of Computer Science, School of Mathematics, Statistics and Computer Science,
University of Tehran, Tehran, Iran
{iranzad}@sadi.ut.ir, {masoudnia; xeraghchi; nowzari}@ut.ac.ir

[2]Electrical and Computer Engineering Department Shahid Rajaee University, Tehran, Iran
ebrahimpour@ipm.ir

Abstract: **In this paper the effect of diversity caused by Negative Correlation Learning (NCL) in the combination of neural classifier is investigated and an efficient way to improve combining performance is presented. Decision Templates and Averaging, as two non-trainable combining methods and Stacked Generalization as a trainable combiner are selected as base ensemble learner and NCL version of them are compared with them in our experiments. Utilizing NCL for diversifying the base classifiers leads to significantly better results in all employed combining methods. Experimental results on five datasets from UCI Machine Learning Repository indicate that by employing NCL, the performance of the ensemble structure can be more favorable compared to that of an ensemble use independent base classifiers.**

Keywords: Neural Networks, Combining Classifiers, Negative Correlation Learning**.**

I. Introduction

Combining multiple classifiers is a machine learning method that uses a set of base classifiers and combine the output of them for producing the final output. This method is known under different names in the literature such as: classifier ensembles, fusion of learners, mixtures of experts, multiple classifier systems, etc. In general point of view, classifier ensembles system is a data classification method made up of an ensemble of base classifiers whose outputs on an input sample data are combined in some way to get a final output on its classification task [1, 2, 3]. Classifier ensembles system may generate more accurate output than each of the base classifiers. It has been proved that Classifier ensembles system is a suitable way to improve the data classification performance, particularly for complex problems such as those involving limited number of sample patterns, high-dimensional feature sets, and highly overlapped classes [4-8]. Combination of multiple classifiers' decisions is a promising solution to gain an acceptable classification performance [9-12]. By utilizing a proper strategy for the construction of an ensemble network, it can be successfully applied to classification problems with imprecise and uncertain information.

An artificial neural network (ANN), commonly called neural network (NN), is one of the most common classifiers which are used in multiple classifier systems. A NN is an information processing system that is inspired by biological nervous systems, and approximates the operation of the human brain, i.e. a NN is a mathematical and computational model that tries to simulate the structure, functional aspects, and behavior of biological neural networks of the human. Neural network is the major technique used in decision making and classification phase for problem solving in the recent years [13-16]. A NN is able to learning and generalizing from some samples and tries to produce meaningful solutions to data which is never seen by it. It can solve the problems even when input data of problem contain errors and are incomplete. In most cases, a NN is an adaptive system that changes its topology based on external or internal information that flows through the network during the training phase. Neural networks are in the class of non-linear statistical system modeling machines. They can be used to model complex relationships between inputs data and outputs data or to find patterns and signals in input data. A neural network may be trained to perform classification, estimation, approximation, simulation, and prediction, i.e. a NN can be configure for a specific application, such as pattern finding, data classification, function approximation through the learning process.

Neural network ensembles methods have two major components, a method to create base NN experts and a method for combining output of base NN experts [17]. Both theoretical and experimental studies [18, 19] are shown that combining procedure is the most effective when the experts' estimates are negatively correlated; but this method is moderately effective when the experts are uncorrelated and only mildly effective when the experts are positively correlated. Therefore, more improved generalization ability can be obtained by combining the outputs of NN experts which are accurate and their errors are negatively correlated [20].

A large number of combining schemes for ensembles classifier exists. As a general viewpoint, there are two types of combination strategy: classifier selection and classifier fusion [21]. The presumption in classifier selection is that each classifier is "an expert" in some local area of feature space; and one or more classifier are selected to assign the label of the input sample in this area. Recall from [21], classifier fusion assumes that all classifiers trained over the whole feature space, and are thereby considers as competitive rather than complementary. Classifier selection has not attracted as much attention as classifier fusion. However, classifier selection is probably the better of the two strategies, if trained well. There also exist combinational schemes between the two pure strategies [22]. Such a scheme; for example, is taking the average of outputs with coefficients which depend on the input x. Thus, the local competence of the classifiers, with respect to x, is measured by the weights, and, more than one classifier is responsible for x and the outputs of all responsible classifiers are fused. The mixture of experts combining method is an example of a method between selection and fusion [21, 23].

On the other point of view, some combining methods do not need training after the classifiers in the ensemble have been trained individually. The Averaging combiner [24] and Decision Template (DT) [25] are two examples of this group. Other combiners need additional training, for example, the weighted average combiner and Stacked Generalization (SG) [26]. The first group is called non-trainable combining and the second one is named trainable [7, 21].

The Averaging method is a relatively simple method of combining models in non-trainable group. In this method, each classifier is trained and outputs for each class are combined by averaging. Commonly, the final result is obtained by just averaging of the estimated posterior probabilities of each base classifier. This simple method gives very good results for some problems [24]. This result is slightly surprising, especially considering this fact that the posterior probabilities averaging are not based on some theoretical support.

DTs are another non-trainable combining technique in fusion category. A DT is a robust classifier ensemble scheme that the output of classifiers is combined by comparing them to a characteristic template for each class. DT ensemble method uses outputs of all classifier to calculate the final output for each class, that is in high contrast to most other ensemble methods which use only the support for that particular class to make their decision [27].

SG is trainable combiner in fusion category. SG is a way of combining multiple classifiers that have been learned for a classification task. In this method, the output pattern of an ensemble of trained experts serves as an input to a second-level expert. The first step is to collect the output of each first level classifier, into a new set. For each instance in the original training set, this data set represents every classifier's prediction. The new data is treated as the input for another classifier and in the second step a learning algorithm is employed to solve this problem [26].

In ensemble research, it is widely believed that the success of ensemble algorithms depends on both the accuracy and diversity among individual learners in the ensemble, demonstrated by theoretical [28, 29] and empirical studies [30]. In general, the component learners in an ensemble are designed to be accurate yet diverse. The empirical results show that the performance of an ensemble algorithm is related with the diversity among base classifiers in the ensemble, and better performance may be achieved with more diversity [31-36]. Many related research on analysis and applications of diversity have been conducted [37-40].

There are various approaches to construct diverse base classifiers. Different learning algorithms, different representation of patterns and partitioning the training set are some of these approaches. Negative Correlation Learning (NCL) is a method based on different learning approach to make diversity between base neural classifiers [30, 39, 41]. This method adds a penalty term to the error function which helps in making the base classifiers as different from each other as possible while encouraging the accuracy of base classifiers.

In this paper, the effect of diversity made by NCL on the combination methods of neural networks is investigated. Some fusion methods which include DTs, SG and average are employed as combining method. The experimental results show that diversified base classifiers by NCL enhanced the performance of all mentioned combining methods.

II. Combining Methods

Learning is a process, that different procedures exceeds to different performance also feature extraction methods influence on performance. There is no standard algorithm to do best performance with a few numbers of data. Each of classification method dependent on different biases performs different from the other. The combination of multiple classifiers was shown to be suitable way to improve the performance of prediction in difficult class determination problems [24]. It has become clear that for more datasets that are complicated, various types of combining rules can improve the final classification. Complexity in classification can be represented from limitation in number of data, classes overlapping and credible noise in data. Results from [18] show that when classifiers have small error rate and are independent in decision-making, combination of classifiers is useful. There are many ways to combine the results of a set of classifiers, depending on the type of the classifiers' output [25]. Classifiers are different by one of these methods. In this section some famous fusion methods which are used in this paper, are described.

A. Decision Templates

DT is a fusion approach taken from fuzzy templates [42]. The general schema of a DT ensemble is shown in Figure 1. Assume that $\{D_1, D_2, ..., D_L\}$ is a set of L classifiers that classify samples set $S = \{(x_1, t_1), (x_2, t_2), ..., (x_N, t_N)\}$, into C classes. The outputs of these classifiers for input x can be arranged as a $L\times C$ matrix called decision profile (DP). The structure of this matrix is defined as follows:

$$DP(x) = \begin{bmatrix} d_{1,1} & \cdots & d_{1,C} \\ \vdots & \ddots & \vdots \\ d_{L,1} & \cdots & d_{L,C} \end{bmatrix} \qquad (1)$$

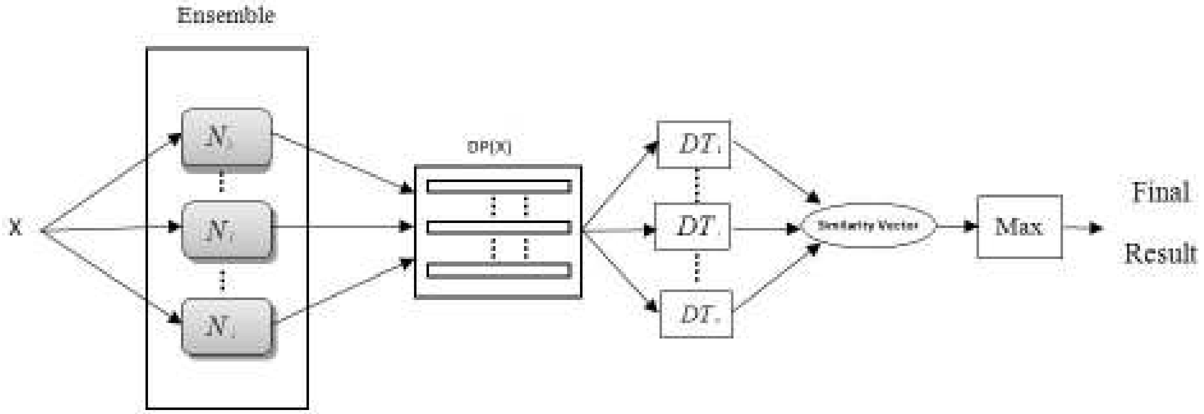

Figure 1. Decision Template schema.

In this matrix, the row i shows the output of classifier D_i and column j shows support of classifiers $D_1, ..., D_L$ for class j, therefore entry $d_{i,j}$ indicates the support of classifier i for class j. Decision template matrix of a class is the average of decision profiles obtained from the training samples belonging to that class. The (k, s)-th element of the DT matrix for class j is calculated by:

$$dt_j(k,s)(x) = \frac{\sum_{q=1}^{N} Ind(x_q, j) d_{k,s}(x_q)}{\sum_{q=1}^{N} Ind(x_q, j)}, \quad (2)$$

where x_q is the q-th element of training dataset, $Ind(x_q,j)$ is an indicator function and its value is 1 if x_q belongs to class j, and 0 otherwise.

During the testing phase, DT scheme compares the DP of the input sample with all of the DTs and suggest a support for each class equal to the similarity between its DT and the DP. There are various similarity measures that can be applied to estimate the similarity. The below function is used as similarity measure in this paper.

$$H(DT_i, DP(x)) = 1 - \frac{1}{LC}\sum_{k=1}^{L}\sum_{s=1}^{C}\sqrt{| DT_i(k,s) - d_{(k,s)}(x) |}. \quad (3)$$

B. Stacked Generalization

In stacked generalization method, the output patterns of some ensemble of trained base experts serve as an input to a second-level expert. The general framework of this method consists of two levels (see Figure 2). The first level, level-0, is formed by base classifiers which are trained using the input data and the target output data. The output of level-0 is then used as the input data to level-1. As is shown in Figure 2, for training set $S = \{(x_1,t_1),(x_2,t_2),...,(x_N,t_N)\}$, a set of K "level-0" base neural networks from N_1^0 to N_1^0 is arranged as the first layer, and is trained by S. Then, the outputs of first layer networks are combined using a "level-1" network N^1. In the other words, first, the level-0 networks are trained using the input data and the target outputs. Then the outputs of the first layer with the corresponding target class are used to train the level-1network. The training algorithm of this modular ensemble can be summarized as follows.

1) From N samples of dataset S, leave out one test sample, and train each base classifiers of the level-0 on the remaining N-1 samples
2) Produce a prediction output for the test sample. The output pattern y=[y_1, y_2,…,y_K] with the target t of the base classifiers of the level-0, for the test sample, becomes a training sample for the classifier of level-1.
3) Repeat the process of Step 1 and 2, in a leave-one-out methodology. This process produces a training set D with N samples, which is used to train the classifier of level-1.
4) To create final learner system, all of the base classifiers of the level-0 are trained one more time using N samples in S.

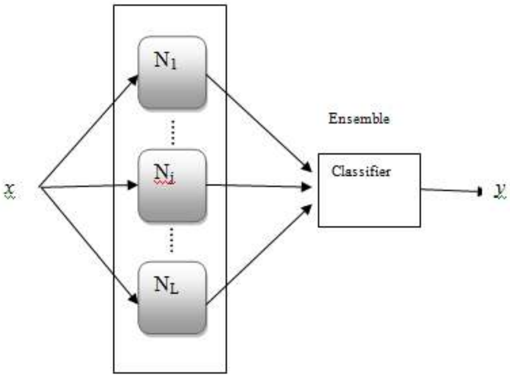

Figure 2. Architecture of Stacked Generalization.

C. Averaging

The Averaging method is a relatively simple method of combining models. In this method, each classifier is trained and outputs for each class are combined by averaging. Once the classifiers in the ensemble are trained they do not require any further training. The Averaging method is used when each classifier produces a confidence estimate (e.g., a posterior). In this case, the winner class is the class with the highest average posterior probability across the ensemble. Let

$S=\{(x_1,t_1),(x_2,t_2),...,(x_N,t_N)\}$ be the training set. At first, the set of classifiers are trained by this set. After that, in the Averaging combination, class label of test data x is obtained by

$$\mu(x)=f(y_1,\cdots,y_L), \qquad (4)$$

where y_i the output of classifier i for test data x, f is the average operation, and $\mu(x)$ is the class label of pattern x.

III. Negative Correlation Learning in combining methods

In this section, at first, a short explanation of the Negative Correlation Learning (NCL) is given. Later ensemble schema with Negative Correlation Learning for designing neural network ensembles is presented.

A. Negative Correlation Learning

Most of the methods for designing neural network ensembles train the individual neural networks independent of each other or in sequential. One of disadvantage of such methods is the loss of interaction among individual neural networks during learning process. It is desired to encourage different individual neural networks to learn different parts of training data set so that the neural network ensemble can learn the whole parts of training data set better. NCL introduced in Section I is a method that supports this aim and trains the ensemble simultaneously and interactively. This method presents a correlation penalty term which has been added to the error function of each individual neural network. During the training process, each individual neural network in the ensemble interacts with other neural networks through their penalty terms in the error functions [30, 43].

Suppose that there is a training set denoted by

$$S=\{(x_1,t_1),(x_2,t_2),...,(x_N,t_N)\},$$

where $x\in R^n$ is the n-dimensional pattern, t is a scalar and the target output and N is the size of the training set. The assumption that the output t_i is a scalar has been made nearly to simplify exposition of idea without loss of generality. NCL considers estimating output by forming an ensemble, whose final output is a simple averaging on outputs of the set of individual neural networks,

$$\overline{y}(n)=\frac{1}{L}\sum_{i=1}^{L}y_i(n), \qquad (5)$$

where L is the number of individual neural networks in the ensemble and $y_i(n)$ is the output of network i on the n-th training pattern, and $\overline{y}(n)$ is the output of ensemble on the same training pattern. The general Architecture of NCL is shown in Figure 3.

The error function E_i for neural network i in negative correlation learning is defined by

$$\begin{aligned}E_i&=\frac{1}{N}\sum_{i=1}^{N}e_i(n)\\&=\frac{1}{N}\sum_{i=1}^{N}\frac{1}{2}(y_i(n)-t_i(n))^2+\lambda\frac{1}{N}.\end{aligned} \qquad (6)$$

Similarly, $t_i(n)$ and $P_i(n)$ are the desired output and correlation penalty function for the n-th pattern respectively. The penalty term $P_i(n)$ is the error function of each individual neural network, and λ is a weighting parameter on the penalty function. The λ parameter controls a trade-off between objective and penalty functions; when λ=0, the penalty function is omitted and we have an ensemble that each neural network is trained independently of the other, using backpropagation algorithm. All networks can train simultaneously and interactively on the same training dataset, and the penalty term has the following form:

$$\begin{aligned}&P_i(n)=(y_i(n)-\overline{y}(n))\sum_{k\neq i}(y_k(n)-\overline{y}(n)),\\&P_i(n)=-(y_i(n)-\overline{y}(n))^2.\end{aligned} \qquad (7)$$

Above equation follows from the fact that

$$\sum_{i=1}^{L}(y_i(n)-\overline{y}(n))=0, \qquad (8)$$

which is based on the fact that the sum of deviations around a mean is equal to zero. Minimization of (7) implies that each neural network has to minimize the difference between the target output and its actual output, like the penalty term. Minimization of the penalty term in (7) results in maximization of the distance between individual neural networks output and the average values. It is clear that from the second part of (7) since there is a negative sign before the distance term in this equation. Therefore the penalty term causes each neural network acts functionally different, and can be expected to get useful diversity among the neural networks. Since the NCL incorporates a diversity measurement directly into the error function, therefore it is considered as an explicit ensemble method [43, 44].

Brown et al. [44] showed that NCL can be viewed as a technique derived from ambiguity decomposition. The ambiguity decomposition, given by Eq. 9, has been widely recognized as one of the most important theoretical results obtained for ensemble learning.

$$(\overline{y}-t)^2=\sum_i w_i(y_i-t)^2-\sum_i w_i(y_i-\overline{y})^2. \qquad (9)$$

This equation states that the Mean-Square-Error (MSE) of the ensemble is guaranteed to be less than or equal to the average MSE of the ensemble members. In this equation t is the target value of an arbitrary data point and $\sum_i w_i=1, w_i\geq 0$ and $\overline{y}$ is the convex combination of the L ensemble members as:

$$\overline{y}=\sum_i w_i y_i. \qquad (10)$$

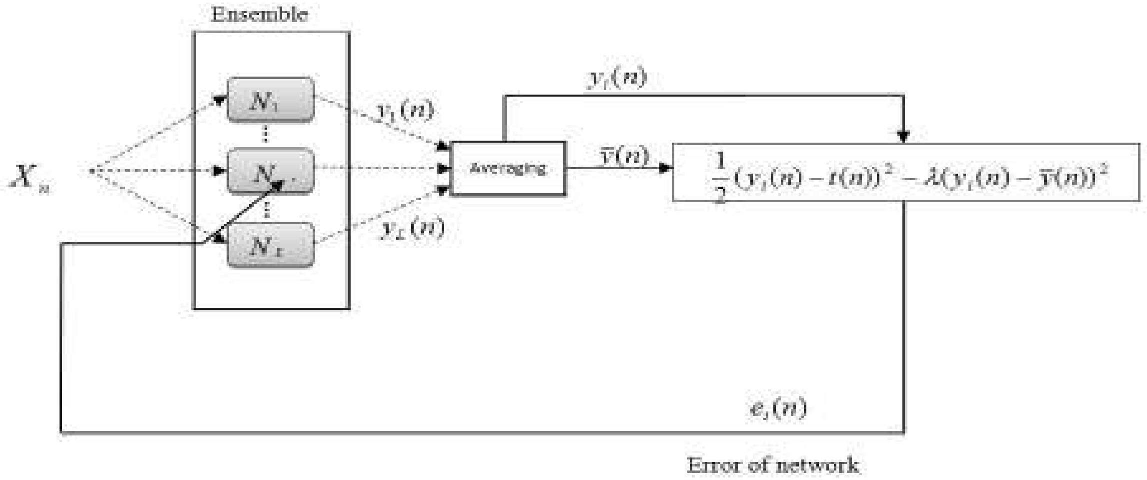

Figure 3. Architecture of Negative Correlation Learning.

The ambiguity decomposition in NCL ensemble method provides a simple expression for the effect of error correlations. The decomposition is composed of two terms. The first term $\sum_i w_i(y_i - t)^2$, is the weighted average error of the individual neural networks. The second term, $\sum_i w_i(y_i - \bar{y})^2$ referred to as the ambiguity, measures the amount of variability among the ensemble members and can be considered to be a correlation of the individual neural networks. Utilizing the coefficient λ allows us to vary the emphasis on the correlation components to yield a near-optimum balance in the trade-off between these two terms; this can be regarded as an accuracy-diversity trade-off [39, 44].

Hansen [28, 45] showed that there is a relationship between bias-variance-covariance decomposition and ambiguity decomposition, in which portions of the first decomposition terms correspond to the portions of the ambiguity decomposition terms. Therefore, any attempt to strike a balance between the two ambiguity decomposition terms leads to three components of the other decomposition that can be balanced against each other, and the MSE tends to a near-minimum condition.

B. Using NCL to diverse base classifiers of combination

Combining classifiers is used as a common way for increasing the classification performance. This technique is helpful, but its effects can be increased by using basic classifiers which have been diversified by a diversity method. This proposed method makes classifiers diverse by NCL and then combine them by some combining methods. It will be shown that this method is successful when Decision Template, Stacked Generalization and Averaging are used as combining methods.

IV. Experimental Results

To evaluate the proposed method, we experimentally compare the Decision Template, Stacked Generalization and Averaging to NCL version of these ensembles. We test the combination rules using real data sets. For ease of comparison, five main benchmark classification datasets are used. These datasets are Sat-image, Vehicle, Pima, Breast and Sonar from the UCI Machine Learning Repository [46]. Information of these data sets is shown in Table 1. Also, a brief review of each data set is given in the following.

Dataset name	Number of Classes	Number of attributes	Training size	Test size
Sat-image	6	4	300	1200
Vehicle	4	18	170	676
Pima	2	8	77	691
Breast	2	30	56	513
Sonar	2	60	42	166

Table 1. Datasets Information.

Sat-image

This data was generated from Land sat Multi-Spectral Scanner image data. It consists of 6435 pixels with 36 attributes. The pixels are crisply classified in 6 classes. 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 %). we used only features # 17 to # 20, as recommended by the database designers.

Vehicle

This data was generated from 2D images of various types of vehicles by application of an ensemble of shape feature extractors to the 2D silhouettes of the objects. It consists of 946 eighteen-dimensional vectors as the extracted features for four types of vehicle: OPEL, SAAB, BUS and VAN. The four classes of this dataset almost have same size.

Pima

This data was from National Institute of Diabetes and Digestive and Kidney Diseases. All considered patients are females at least 21 years old of Pima Indian heritage. The feature vector of each patient information is an eight-dimensional vector, including both the integer and real valued numbers. The dataset consists of 768 cases for two types: healthy people against who have diabetes.

Breast Cancer Wisconsin

This data was computed from a digitized 2D image of a fine needle aspirate of a breast mass. It describes characteristics of the cell nuclei present in the image. The dataset consists of 569 thirty-two dimensional vectors as the real-valued features are computed for each cell nucleus. The two classes of this dataset present two types of diagnosis: malignant and benign tumors.

Sonar

This dataset obtained by bouncing sonar signals off cylinders made up from two different materials, at various angles and under various conditions. It contains 111 patterns obtained by bouncing sonar signals off a metal cylinder and 97 patterns obtained from rocks under similar conditions. The transmitted sonar signal is a frequency-modulated chirp, rising in frequency. The data set contains signals obtained from a variety of different aspect angles, spanning 90 degrees for the cylinder and 180 degrees for the rock. Each pattern is a set of 60 numbers in the range 0.0 to 1.0. Each number represents the energy within a particular frequency band, integrated over a certain period of time. Therefore, this dataset consists of 208 sixty dimensional real value vectors for two groups of bouncing sonar signals off metal and rocks.

In all experiments there are five MLPs in the ensemble. The strength parameter of NCL tested in the interval of [0:0.2:1] and 0.8 has been chosen as the best one and 50 epochs has been done to train the ensemble. The weak classifiers have been chosen as the base classifiers which are obtained by using neural networks with low complexity. The small sub-sample of each dataset is selected randomly as the training set. It is applied to show generalization ability of proposed method. The small part of dataset (about 10%) is selected for training, but for some datasets including: Vehicle and Pima, following this approach was not practical because they have many classes that it is impossible to train well by that size of training data and it have to be chosen more data as training set (around 20%). Also, since Sonar dataset has small instances, classifiers cannot train by 10 percentages of dataset and it has to be chosen more training data (around 20%).

It is expected that this proposed method would have thriving results in the case of small sample size problems, and this is a way for solving that problems which are difficult to classify by simple neural networks like MLP. The related tables of the results are illustrated below.

The results of proposed method in Decision Template as a combining method are presented in Table 2.

Dataset	Average percentages of 5 base classifiers	DT	NCL-DT
Sat-image	45.866(±3.9)	50.0833	61.5
Vehicle	62.071(±2.19)	65.2367	73.6686
Pima	63.412(±1.34)	70.33	75.1085
Breast	60.768(±1.2)	88.889	97.465
Sonar	44.270(±1.46)	63.8554	65.0602

*Table 2***.** Results for combining based on Decision Template.

The results of proposed method in Stacked Generalization method are presented in Table 3.

Dataset	Average percentages of 5 base classifiers	SG	NCL-SG
Sat-image	45.866(±3.9)	81.1667	83.1667
Vehicle	62.071(±2.19)	68.3784	75
Pima	63.412(±1.34)	65.4124	72.2533
Breast	60.768(±1.2)	62.768	98.4405
Sonar	44.270(±1.46)	45.7831	62.0482

*Table 3***.** Results of Stacked Generalization.

The results of proposed method in Averaging combining method are presented in Table 4.

Dataset	Average percentages of 5 base classifiers	Averaging	NCL-Averaging
Sat-image	45.866(±3.9)	60.6994	73.0225
Vehicle	62.071(±2.19)	62.9053	66.0266
Pima	63.412(±1.34)	65.3179	68.0656
Breast	60.768(±1.2)	62.6459	89.1051
Sonar	44.270(±1.46)	45.5090	62.2695

Table 4. Results of combining based on averaging.

It is obvious that NCL improved the performance of all investigated methods considering to the results shown in the tables. The examples of the above tables show that the classifiers on the independent data sets classify a large fraction of the data correctly when the NCL method uses for diversity. The results also show that the independent views of the different classifiers can contribute significantly to correct the output for some of the more difficult data in NCL version of each combining classifiers.

V. Conclusion

The approach introduced in this paper is using diverse neural network classifiers for combining, i.e. the main goal of this work was to investigate the relative effect of NCL over classifier outputs. The proposed method has two steps: at first, the base classifiers are diversified by NCL and then combining methods are applied. Two non-trainable combining methods including: averaging and DTs and a trainable combiner SG are investigated. Experimental results show that diverse base classifiers improve the performance of combining in both trainable and non-trainable combining methods.

References

[1] L. Rokach, 2010, “Ensemble-based classifiers”, *Artificial Intelligence Review,* 33(1), pp. 1-39.

[2] T. Wilk, and M. Wozniak, 2011, “Soft computing methods applied to combination of one class classifiers”, *Neurocomputing*, 75(1), pp. 185-193

[3] S. Bhardwaj, S. Srivastava, J. Gupta, and A. Srivastava, 2012, "Chaotic time series prediction using combination of hidden markov model and neural nets", *International Journal of Computer Information Systems and Industrial Management Applications*, 4, pp. 236-243.

[4] T. P. Tran, T. T. S. Nguyen, P. Tsai et al., 2011, "BSPNN: boosted subspace probabilistic neural network for email security", *Artificial Intelligence Review*, 35(4), pp. 369-382.

[5] S. Kotsiantis, 2011, "An incremental ensemble of classifiers", *Artificial Intelligence Review*, 36(4), pp. 249-266.

[6] S. Kotsiantis, 2011, "Combining bagging, boosting, rotation forest and random subspace methods", *Artificial Intelligence Review*, 35(3), pp. 223-240.

[7] L. Rokach, 2010, *Pattern Classification Using Ensemble Methods*: World Scientific Pub. Co. Inc., Hackensack, NJ.

[8] Z. Yu-Quan, O. Ji-Shun, C. Geng, and Y. Hai-Ping, 2011, "Dynamic weighting ensemble classifiers based on cross-validation", *Neural Computing & Application*, 20(3), pp. 309–317.

[9] M. Tabassian, R. Ghaderi, and R. Ebrahimpour, 2012, "Combination of multiple diverse classifiers using functions for handling data with imperfect labels", *Expert Systems with Applications*, 39, pp. 1698-1707.

[10] M. Tabassian, R. Ghaderi, and R. Ebrahimpour, 2011, "Knitted fabric defect classification for uncertain labels based on Dempster-Shafer theory of evidence", *Expert Systems with Applications*, 38, pp. 5259-5267.

[11] R. Ebrahimpour, E. Kabir, and M. Yousefi, 2011, "Improving mixture of experts for view-independent face recognition using teacher-directed learning", *Machine Vision & Applications,* 22, pp. 421-432.

[12] F. Behjati-Ardakani, F. Khademian, A. Nowzari-Dalini, and R. Ebrahimpour, 2011, "Low resolution face recognition using mixture of experts", *World Academy of Science, Engineering and Technology*, 80, pp. 879-883.

[13] R. Ghazali, N. M. Nawi, and M. Z. M. Salikon, 2009, "Forecasting the UK/EU and JP/UK trading signals using polynomial neural networks", *International Journal of Computer Information Systems and Industrial Management Applications*, 1, pp. 110-117.

[14] P. Klement and V. Snasel, 2010, "SOM neural network - a piece of intelligence in disaster management", *International Journal of Computer Information Systems and Industrial Management Applications*, 2, pp. 243-251.

[15] P. M. Ciarelli, E. Oliveira, C. Badue, and A. F. D. SouzaKlement, 2009, "Multi-Label text categorization using a probabilistic neural Network", *International Journal of Computer Information Systems and Industrial Management Applications*, 1, pp. 133-144.

[16] M. Barakata, F. Druauxa, D. Lefebvrea, M. Khalilb, O. Mustaphac, 2011, "Self adaptive growing neural network classifier for faults detection and diagnosis", *Neurocomputing*, 74(18), pp. 3865-3876.

[17] R. Polikar, 2007, "Bootstrap-inspired techniques in computational intelligence", *IEEE Signal Processing Magazine,* 24(4), pp. 59-72.

[18] K. Tumer and J. Ghosh, 1996, "Error correlation and error reduction in ensemble classifiers", *Connection Science,* 8(3), pp. 385-404.

[19] R. Polikar, 2006, "Ensemble based systems in decision making", *IEEE Circuits and Systems Magazin ,* 6(3), pp. 21-45.

[20] R. A. Jacobs, 1997, "Bias/variance analyses of mixtures-of-experts architectures", *Neural Computation,* 9(2), pp. 369-383.

[21] L. I. Kuncheva, 2004, *Combining Pattern Classifiers, Methods and Algorithms*: John Wiley, Hoboken, NJ.

[22] L. I. Kuncheva, 2002, "Switching between selection and fusion in combining classifiers: An experiment", *IEEE Transactions on Systems Man and Cybernetics Part B-Cybernetics,* 32(2), pp. 146-156.

[23] R. Ebrahimpour, H. Nikoo, S. Masoudnia et al., 2010, "Mixture of MLP-experts for trend forecasting of time series: A case study of the Tehran stock exchange", *International Journal of Forecasting*, 27(3), pp. 804-816.

[24] J. Kittler, 1998, "Combining classifiers: A theoretical framework", *Pattern Analysis and Applications,* 1(1), pp. 18-27.

[25] L. I. Kuncheva, J. C. Bezdek, and R. P. W. Duin, 2001, "Decision templates for multiple classifier fusion: an experimental comparison", *Pattern Recognition,* 34(2), pp. 299-314.

[26] D. H. Wolpert, 1992, "Stacked generalization", *Neural Networks,* 5(2), pp. 241-259.

[27] L. I. Kuncheva, 2001, "Using measures of similarity and inclusion for multiple classifier fusion by decision templates", *Fuzzy Sets and Systems,* 122(3), pp. 401-407.

[28] L. K. Hansen and P. Salamon, 1990, "Neural network ensembles", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 12(10), pp. 993-1001.

[29] A. Krogh and J. Vedelsby, 1995, "Neural network ensembles, cross validation and active learning", In *Proceedings of Advances in Neural Information Processing Systems 7*, G. Teasuro, D. S. Touretzky, and T. K. Leen (eds.), MIT Press, Cambridge, MA, pp. 231–238.

[30] Y. Liu and X. Yao, 1999, "Ensemble learning via negative correlation", *Neural Networks,* 12(10), pp. 1399-1404.

[31] P. Cunningham and J. Carney, 2000, "Diversity versus quality in classification ensembles based on feature selection", In *Proceedings of 11th European Conference on Machine Learning*, pp. 109-116.

[32] L. I. Kuncheva and C. J. Whitaker, 2003, "Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy", *Machine Learning,* 51(2), pp. 181-207.

[33] T. Windeatt, 2006, "Accuracy/diversity and ensemble MLP classifier design", *IEEE Transactions on Neural Networks ,* 17(5), pp. 1194-1211.

[34] G. I. Webb and Z. Zheng, 2004, "Multistrategy ensemble learning: Reducing error by combining ensemble learning techniques", *IEEE Transactions on Knowledge and Data Engineering,* 16(8), pp. 980-991.

[35] Z. H. Zhou and Y. Yu, 2005, "Ensembling local learners through multimodal perturbation", *IEEE Transactions*

on Systems Man and Cybernetics Part B-Cybernetics, 35(4), pp. 725-735.

[36] L. I. Kuncheva, 2005, “Using diversity measures for generating error-correcting output codes in classifier ensembles”, *Pattern Recognition Letters,* 26(1), pp. 83-90.

[37] R. Kohavi and D. H. Wolpert, 1996, “Bias plus variance decomposition for zero-one loss functions”, In *Proceedings of the 13th International Conference on Machine Learning,* pp. 275-283.

[38] D. Partridge and W. Krzanowski, 1997, “Software diversity: practical statistics for its measurement and exploitation”, *Information and software technology,* 39(10), pp. 707-717.

[39] H. Chen and Xin Yao, 2008, “Regularized negative correlation learning for neural network ensembles”, *IEEE Transactions on Neural Networks*, 20(12), pp.1962-1979.

[40] J. J. Rodriguez and L. I. Kuncheva, 2006, “Rotation forest: A new classifier ensemble method”, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 28(10), pp. 1619-1630.

[41] Y. Liu and X. Yao, 1999, “Simultaneous training of negatively correlated neural networks in an ensemble”, *IEEE Transactions on Systems Man and Cybernetics Part B-Cybernetics,* 29(6), pp. 716-725.

[42] K. Woods, W. P. Kegelmeyer, and K. Bowyer, 1997, “Combination of multiple classifiers using local accuracy estimates”, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 19(4), pp. 405-410.

[43] M. F. Amin, M. M. Islam, and K. Murase, 2009, “Ensemble of single-layered complex-valued neural networks for classification tasks”, *Neurocomputing,* 72(10), pp. 2227-2234.

[44] G. Brown, and J. M. Wyatt, 2003, “Negative correlation learning and the ambiguity family of ensemble methods”, *Lecture Note in Computer Science,* 2709, pp. 266-275.

[45] J. V. Hansen, 1999, “Combining predictors: comparison of five meta machine learning methods”, *Information Sciences,* 119(1), pp. 91-105.

[46] A. Asuncion and D. J. Newman, 2010, “UCI machine learning repository [http://www.ics.uci.edu/ MLRepository.html]”, University of California, School of Information and Computer Science, Irvine, CA.

Author Biographies

Abbas Nowzari-Dalini received his PhD degree from the School of Mathematics, Statistics, and Computer Science in 2003. He is an associate professor of School of Mathematics and Computer Science, University of Tehran, and is currently the director of Graduate studies in this school. His research interests include neural networks, bioinformatics, combinatorial algorithm, parallel algorithms, DNA computing, and computer networks.

Reza Ebrahimpour received the BS degree in electronics engineering from Mazandaran University, Mazandaran and the MS degree in biomedical engineering from Tarbiat Modarres University, Tehran, Iran, in 1999 and 2001, respectively. He received his PhD degree in July 2007 from the School of Cognitive Science, Institute for Studies on Theoretical Physics and Mathematics, where he worked on view-independent face recognition with Mixture of Experts. He is an assistant professor of Shahid Rajaee University. His research interests include human and machine vision, neural networks, and pattern recognition.

Arash Iranzad received the BS degree in computer science from School of Mathematics, Statistics, and Computer Science, University of Tehran in 2011. He is currently a MS student in Computer Science Department, Brock University, Ontario, CA. His research interests are neural networks, pattern recognition, multiple classifier systems, and machine vision.

Saeed Masoudnia received the MS degree in computer science from School of Mathematics, Statistics, and Computer Science, University of Tehran in 2011. His research interests are neural networks, pattern recognition, machine learning, computational neuro-science and models of human vision.

Fatemeh Cheraghchi received the MS degree in computer science from School of Mathematics, Statistics, and Computer Science, University of Tehran in 2010. His research interests are neural networks, pattern recognition, multiple classifier systems, quantum computing and analysis of algorithms.