International Journal of the Physical Sciences Vol. 6(30), pp. 7007 - 7015, 23 November, 2011
Available online at http://www.academicjournals.org/IJPS
DOI: 10.5897/IJPS11.1069
ISSN 1992 - 1950 ©2011 Academic Journals

Full Length Research Paper

Persian handwritten digits recognition: A divide and conquer approach based on mixture of MLP experts

Mohammad Masoud Javidi[1]*, Reza Ebrahimpour[2] and Fatemeh Sharifizadeh[3]

[1]Department of Computer Science, Shahid Bahonar University of Kerman, Kerman, Iran.
[2]Brain and Intelligent Systems Research Laboratory, Department of Electrical and Computer Engineering, Shahid Rajaee Teacher Training University, P. O. Box: 16785-163, Tehran, Iran.
[3]Department of Computer Science, Shahid Bahonar University of Kerman, Kerman, Iran.

Accepted 12 October, 2011

In pursuit of Persian handwritten digit recognition, many machine learning techniques have been utilized. Mixture of experts (MOE) is one of the most popular and interesting combining methods which has great potential to improve performance in machine learning. In MOE, during a competitive learning process, the gating networks supervise dividing input space between experts and experts obtain specialization on those subspaces. In this model, simple linear networks are used to form the building blocks of the MOE architecture. But in this study, due to complexity of Persian handwritten digit classification, the multi layer perceptrons, MLP, is used as gating and expert networks. We call this architecture the mixture of multilayer perceptrons experts (MOME). Comparative evaluation is accomplished with two real-world datasets: SRU Persian numerals and a very large dataset of Persian handwritten digit (HODA). In this paper, experiments are conducted to evaluate the performance of MOME with various appraisal criteria and also, classification capabilities of various neural network ensembles are compared with MOME. Our experimental results indicate significant improvement in recognition rate of our investigated method, MOME, in all practical tests.

Key words: Mixture of experts, hand written digit recognition, combining classifiers, mixture of multi-layer perceptrons experts.

INTRODUCTION

In the last few decades, numerous methods have been proposed for machine recognition of handwritten characters, especially for the more popular languages such as English, Japanese and Chinese. In particular, handwritten numeral recognition has attracted much attention, and various techniques (pre-processing, feature extraction, and classification) have been proposed (Liu et al., 2003; Trier et al., 1996; Ho et al., 1994; Xu et al., 1991; Suen et al., 1990).

In contrast, very little research had reported the recognition of Persian (Arabic) handwritten digits (Liu et al., 2009; Pan et al., 2009; Borji et al., 2008; Suen et al., 2006; Soltanzadeh and Rahmati, 2004; Amin, 1998). However, today research on Farsi (Persian) scripts and numerals is receiving increasing attention because of the automatic processing of handwritten data.

Combining classifiers is an approach which improves the performance of classification, particularly for complex problems such as those involving a limited number of patterns, high-dimensional feature sets, and overlapping classes (Ho et al., 1994; Ghaderi, 2000).

There are two main strategies for combining classifiers: fusion and selection (Kuncheva, 2004). In fusion, we suppose that each ensemble member is trained on the whole feature space (Xu et al., 1992; Ng and Abramson, 1992; Kittler et al., 1998), whereas in selection, each member is assigned to learn a part of the feature space (Woods et al., 1997; Jacobs et al., 1991; Alpaydin and Jordan, 1996).

Thus, in the former strategy, the final decision is made on the basis of the decisions of all members, while in the latter strategy, the final decision is made by aggregating the decisions of one or some of the experts (Kuncheva,

*Corresponding author. E-mail: javidi@mail.uk.ac.ir. Tel: +98 341 320 2251.

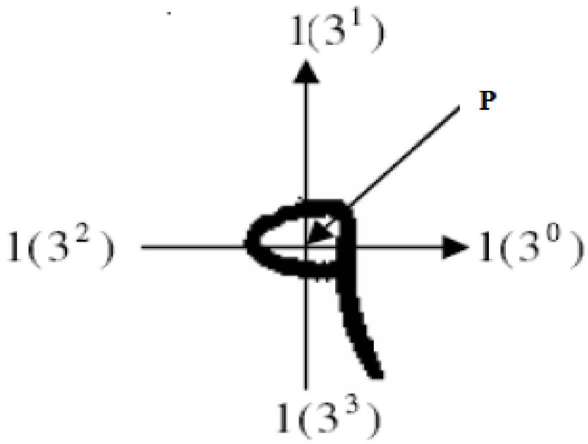

Figure 1. Locus number of point P is $(1111)_3 = 1\times 3^3 + 1\times 3^2 + 1\times 3^1 + 1\times 3^0 = (40)_{10}$.

2004; Haykin, 1998). Classifier selection is probably the better of the two strategies, if the members are trained well (Kuncheva, 2004). Combining classifiers based on the fusion of outputs of a set of different classifiers have been developed as a method for improving the performance of systems (Tax et al., 2000; Liu, 2005). Classifier fusion is categorized into two classes, trainable and non trainable. Ebrahimpour and Sharifizadeh (2009) used static methods to Persian handwritten digit recognition.

One of the most popular methods of classifier selection is the mixture of experts (MOE), originally proposed by Jacobs et al. (1991) and which falls under the category of classifier selection. In MOE, the problem space is divided into several subspaces, and the outputs of the experts are combined by a gating network. In other words, a gating network implements competitive selection between a numbers of simple homogeneous modules (experts).

In this paper, a method for Persian handwritten digit recognition, based on “mixture of experts” (MOE) is investigated. In Persian handwritten digit recognition, each expert must localize on a difficult subproblem. Thus, we use a modified structure for the experts and gating network. In this architecture, we use MLP instead of linear networks as experts together with a gating network. We call this architecture the Mixture of Multilayer Perceptrons Experts, MOME. In order to match the gating and expert networks, to endow the model the ability to select the expert network best at solving the problem, a revised learning algorithm is presented.

FEATURE EXTRACTION

The selection of a feature extraction method with good discriminating power is probably the single most important stage for transforming the input space into the feature space. In our experiments to avoid a high dimensional and redundant input space and optimally design and train the experts, we first use the characteristic loci method and then principle component analysis (PCA). These methods are described following.

Characteristic loci

In the first stage of our proposed model, we use characteristic loci as a robust feature extraction method discussed in the literature of Persian handwritten digit recognition. In this method, usually vertical and horizontal vectors are generated (Glucksman, 1967; Ebrahimi and Kabir, 2008; Knoll, 1969) and then a number is assigned to each pixel of image. These numbers are used in computing feature vectors. The numbers are dependent on the number of times the line segments in right, upward, left and downward directions intersect with the character body. In this application, the maximum number of intersections is limited to 2, since the shape of Persian digits can be described by this limitation. Thus, for each pixel of image, a four digit number of base 3 is attained. These numbers are called locus numbers and are between 0 and 80. Figure 1 exhibits an example of calculating the locus number at the point P.

This is done for all pixels of image. Thus, the feature vector has 81 components and each element of this vector depicts the sum of background pixels that have a locus number corresponding to that element. Features are normalized by dividing them by the total number of background image.

Principle component analysis (PCA)

After using the characteristic loci feature extraction method, we use principle component analysis, PCA, to avoid a high dimensional and redundant input space and optimally design and train the experts.

PCA is a useful statistical technique that has found application in fields such as image compression, image processing and image recognition and is a common technique for extracting informative low dimensional patterns in data of high dimension, with no harmful loss of information content. It is basically a way of identifying patterns in data, along with their similarities and differences (Martinez and Kak, 2001; Aradhya et al., 2008).

The PCA method is implemented in the following four steps: Normalizing the data, calculating the covariance matrix, calculating the eigenvectors and eigenvalues of the covariance matrix and finally, choosing components and forming a feature vector.

BRIEF DESCRIPTION OF THE USED CLASSIFIER FUSION METHODS

Here, we are going to present a brief description of each of the

$$DP(x)=\begin{bmatrix} d_{11}(x) & \dots & d_{1j}(x) & \dots & d_{1c(x)} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ d_{i1}(x) & \cdots & d_{ij}(x) & \cdots & d_{ic}(x) \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ d_{L1}(x) & \cdots & d_{Lj}(x) & \cdots & d_{Lc}(x) \end{bmatrix}$$

Figure 2. Decision profile matrix for an input pattern x. Each row in this matrix is the output of classifier $D_i(x)$ and each column exhibits the supports from classifier $D_1,\dots,D_L$.

used fusion techniques. In the subsequently in the experimental results, the accuracy of these methods is compared with the result of our method.

Suppose that $x\in\Re^n$ be a feature vector, and $\{D_1,\dots,D_L\}$ be a set of classifiers and $\Omega=\{\omega_1,\dots,\omega_c\}$ be the set of class labels. We denote the output of the $i'th$ classifier as $D_i(x)=[d_{i,1}(x),\dots,d_{i,c}(x)]^T$, where $d_{i,j}(x)$ indicated the support that classifier D_i gives to the supposition that x comes from class ω_j.

The L classifier outputs for an input pattern x can be arranged in a decision profile matrix $(DP(x))$ as shown in the Figure 2 (Kuncheva, 2004).

There are two general approaches to use $DP(x)$ to find label of the input x : *Class-indifferent* and *class-conscious* methods.

In this paper, we appraise our method, which is under the category of dynamic structure with most popular fusion method. Thus we compare the performance of our method with decision template combining method and four of the most famous combining methods under the category of class conscious (namely minimum, maximum, product and average). These methods are briefly described in the following.

Decision templates

One of the most popular methods of class-indifferent is decision templates, DT, originally proposed by Kuncheva et al. (2001). DT model was developed based on a set of c matrices called *decision templates* (DT). DT is a robust classifier fusion scheme that combines classifier outputs by comparing them to a characteristic template for each class. DT fusion uses *all* classifier outputs to calculate the final support for each class, which is in sharp contrast to most other fusion methods that use *only the support for that particular class* to make their decision (Ebrahimpour and Sharifizadeh, 2009).

Minimum rule

In this method, the output node that is the maximum value among the minimums of experts' outputs, determines the final decision.

Maximum rule

In this method, the output node that is the maximum value among the maximums of experts' outputs, determines the final decision.

Average method

In this method, the final decision is made by averaging the experts' outputs.

Product method

In this method, the output node that is the maximum value among the multiplication of experts' outputs, determines the final decision.

INVESTIGATED METHOD: MOME

A mixture of experts is the most famous method in the category of dynamic structures of classifier combining. In this method, the input signal is directly involved in actuating the mechanism that integrates the outputs of the individual experts into an overall output (Haykin, 1998). The mixture of experts is composed of expert networks and a gating network. The experts compete to learn the training patterns and the gating network mediates the competition. The gating network is simultaneously trained to combine the experts' outputs. In this modular neural network, the learning process proceeds by fusing self organized and supervised forms of learning. The experts are technically performing supervised learning in that their individual outputs are combined to model the desired response. There is however, a sense in which the experts are also performing self–organized learning; that is they self–organize to find a good partitioning of the input space so that each expert does well at modeling its own subspace, and so, as a whole group, they also model the input space well. The learning algorithm of the mixture structure is described in Jacobs et al. (1991).

In Jacob's model, simple linear networks are used to form the building blocks of the MOE architecture, in which the gating networks, according to spatial similarity, divide the input space into subproblems and each expert is a specialist for its subproblem. However, in a complex problem, it is required to use stronger experts. In Persian handwritten digit recognition, each expert must localize on a difficult subproblem. So in our model, in order to improve the performance of the expert networks, and consequently the handwritten digit recognition accuracy, we use a revised version of MOE in which MLP instead of linear networks or experts are used. We call this model a mixture of multilayer perceptron experts (MOME). The number of experts in the MOME network can vary, but for understanding the working of the model shall here suppose that five experts are used. A sketch of model is shown in Figure 3.

The gating network's learning rules attempt to maximize the likelihood of the training set assuming a Gaussian mixture model in which each expert is responsible for one component of the mixture. Thus, the network itself partitions the input space and hence we call it a self-directed partitioning network. The experts are directed towards different subspaces and according to their field expertise. The detailed learning process as follows.

In order to match the gating and expert networks and to endow the model with the ability to select the best expert network for solving the problem, the learning algorithm is corrected by an estimation of the posterior probability of the generation of the desired output by each expert. Using this new learning method, the MLP expert networks' weights are updated and the procedure is

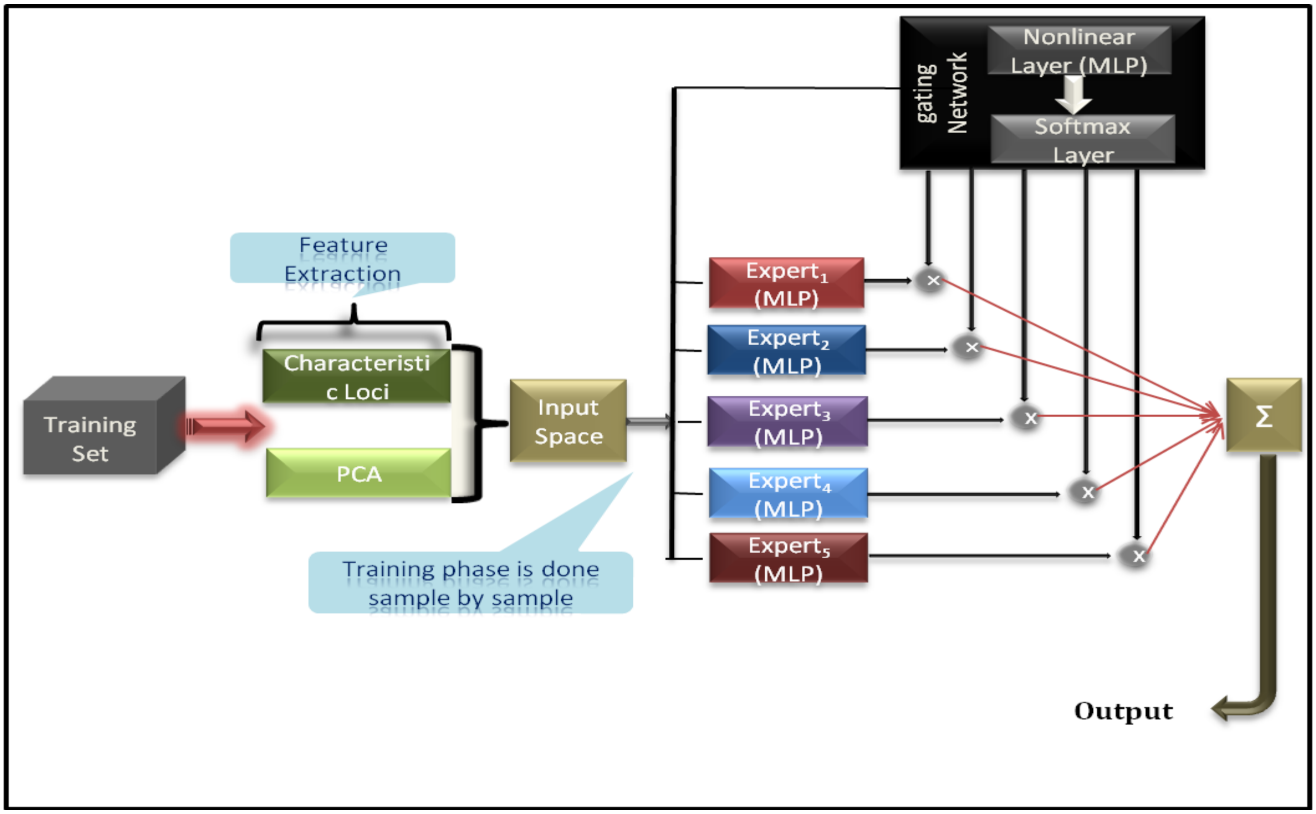

Figure 3. Sketch of the MOME models.

repeated for the training dataset.

Each expert is an MLP network with one hidden layer that computes an output vector O_i as a function of the input stimuli vector, x, and a set of parameters such as weights of hidden and output layers and a sigmoid function activation function. It is assumed that each expert specializes in a different area of the input space. The gating network assigns a weight g_i to each of the experts' output O_i. In fact g_i is a function of the input vector x and a set of parameters such as weights of its hidden and output layers and a sigmoid activation. The g_i can be interpreted as estimates of the prior probability which expert i can generate the desired output y. The gating network is composed of two layers: The first layer is an MLP network, and the second layer is a softmax nonlinear operator. Thus the gating network computes O_g, the output of the MLP layer of the gating network, and then applies the softmax function to get:

$$g_i = \frac{\exp(O_{gi})}{\sum_{i=1}^{N} \exp(O_{gj})}, i = 1, \ldots, 5 \tag{1}$$

The g_i are nonnegative and sum to 1. The final mixed output of the entire network is

$$O_T = \sum_{i=1}^{5} O_i g_i \tag{2}$$

The "normalized" exponential transformation of Equation (1) may be viewed as a multi–input generalization of the logistic function. It preserves the rank order of its input values, and is a differentiable generalization of the "winner-takes-all" operation of picking the maximum value. For this reason, the function of Equation (1) is referred to as softmax. W_h and W_y are the weights of input in the hidden layer of experts and hidden to output layer of experts, respectively. W_{hg} and W_{yg} are the weights of input to hidden layer of the gating network and hidden to output layer of gating network, correspondingly. These weights are learned using the back-propagation, BP, algorithm. Assuming the probability density associated with each expert is Gaussian with identity covariance matrix, for each expert i, the MLP obtains the following online learning rules:

$$\Delta W_y = \eta_e h_i (y - O_i)(O_i(1 - O_i))O_{hi}^T \tag{3}$$

$$\Delta W_h = \eta_e h_i W_y^T (y - O_i)(O_i(1 - O_i))O_{hi}(1 - O_{hi})x_i \tag{4}$$

Figure 4a. Samples of Farsi numerals from training set.

Figure 4b. Samples of Farsi numerals from testing set.

$$\Delta W_{yg} = \eta_g (h-g)\big(O_g(1-O_g)\big)O_{hg}^T \tag{5}$$

$$\Delta W_{hg} = \eta_g W_{yg}^T (h-g)\big(O_g(1-O_g)\big)O_{hg}(1-O_{hg})x_i \tag{6}$$

where η_e and η_g are learning rates for the experts and the gating network, respectively, O_{hi} is the output of the expert network's hidden layer, O_{hg} is the output of the hidden layer of gating network, and h_i is an estimate of the posterior probability that expert i can generate the desired output , y :

$$h_i = \frac{g_i \exp(\frac{-1}{2}(y-O_i)^T(y-O_i))}{\sum_j g_j \exp(\frac{-1}{2}(y-O_j)^T(y-O_j))} \tag{7}$$

This can be thought of as a softmax function computed on the inverse of the sum of squared errors of each expert's output, smoothed by the gating network's current estimate of the prior probability that the input pattern was drawn from expert i s area of specialization. As the network's learning process progresses, the expert networks "compete" for each input pattern, while the gating network rewards the winner of each competition with stronger error feedback signals. Thus, over time, the gate partitions the input space in response to the expert's performance.

EXPERIMENTAL RESULTS

To evaluate the performance of our proposed method and also exhibit the advantage of using it in recognition of Persian handwritten digits, we carried out several experiments on two datasets.

Datasets

We evaluate our recognition methods on two datasets: SRU (Shahid Rajaee University) Persian numerals and a very large dataset of Persian handwritten digits (HODA) (Khosravi and Kabir, 2007). Each database is divided into training, validation, and test sets, which includes approximately 58, 17, and 25% of the available data, respectively.

The SRU database

For the training and testing phases, we collected 8600 digit images written by 860 different people, each person writing down each of the 10 digits. The participants were selected among the undergraduate students from universities in Tehran, Iran. 5000 samples were used for training and 1450 samples were used for validation and 2150 samples for testing. All of the samples were scanned at 300 dpi resolution in the grayscale format. The images sizes were 40 × 40 pixels. Two samples of 10 classes for training set and testing set are shown in Figure 4.

In both Figure 4a and b, first row exhibits the images that are recognized by humans without any ambiguity. Images in this category are clear and unambiguous. They have all the necessary structural primitives and have typical connectivity of the primitives. The second row presents images that humans have difficulty in identifying because of noise, filled loops, cursive writing, over-segmentation or similarity of their primitives and structures, etc. This dataset is available at http://bislab.ir/OCR/.

The HODA dataset

Khosravi and Kabir (2007) have introduced a very large corpus of Farsi handwritten digits. Binary images of 102,352 digits were extracted from about 12,000 registration forms of two types, filled by bachelor and senior high school students. These forms were scanned at 200 dpi with a high speed scanner. The preprocessing, finding areas of interest and digit extraction were performed and this Farsi digit dataset is divided into a set of 60,000 samples used for the training set and a set of 20,000 samples for testing.

The samples in this dataset are very accurate and simple, because the registration forms were scrupulously filled for the university entrance examinations and students pay great attention when completing such forms. Thus to provide benchmark for evaluating our method for Persian handwritten digit recognition, we extract harder samples from HODA database. So using K-nearest neighbors method with K = 6, we selected the hard data which were classified into more than three classes. Some samples of 10 classes are shown in Figure 5. So finally we ended up with following subset for our experiments:

MOME vs. a single MLP

To evaluate the performance of MOME, we compared it with a single MLP on the both datasets. After using characteristic loci feature extraction method, in order to decrease computational load and to achieve high accuracy, dimensionality reduction was performed using

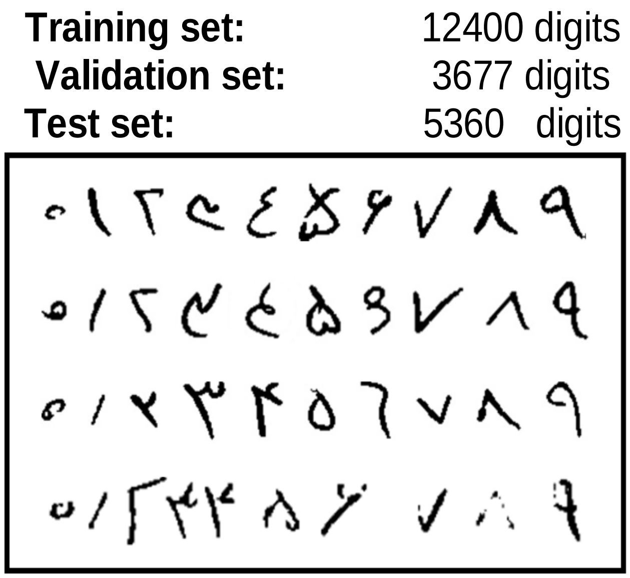

Figure 5. Samples of the subset of HODA Farsi dataset.

PCA. In the first stage, the number of PCA components must be specified. We use a MLP with 95 hidden neurons and 10 output nodes to specify the number of PCA components. Table 1 shows the error rate of the MLP computed with different number of PCA components for the validation sets of SRU and subsets of the HODA dataset.

A 30-dimensional subspace was found to be optimal for the SRU dataset and 50-dimensional subspace was the best for the subset of the HODA dataset. Here these global eigenspaces were used in all subsequent experiments.

The MLP has learning parameters, such as number of hidden neurons, number of epochs and the learning rate. To find the best parameter values, we adjusted the parameters on the training set, and tested them on the validation set. Parameters that gave the best results on the validation set were used for classifying on the testing set.

To take a decision about the topology of a single MLP, different topologies were examined. It was found that, the MLP with the structure of 30:100:10 was the best for the SRU dataset, and the structure of 50:95:10 was the best for the subset of the HODA dataset. In this experiment, the learning rate for the MLP was 0.1 on the SRU dataset and 0.25 on the subset of the HODA dataset.

To find the best structure for the experts and gating network of MOME, different topologies were examined on the both datasets. The results of this experiment are reported in Table 2a and b, where, for a variety of the number of hidden neurons for gating and expert networks, the error rate on the validation sets are listed. We found that the optimum values for η_e and η_g were 0.09 and 0.03, respectively, for the SRU dataset and 0.1 and 0.05, respectively, for the subset of the HODA and we compare its result with fusion methods so we segregate between structure of experts and gating in dataset. Error rates of different topologies of MOME on the used datasets are reported in Table 2a and b. In each column, for fixed values of hidden neurons of gating network and experts, error rate which is the average of ten runs with different random initial weights are listed.

As shown in Table 2a and b, the best structure for the expert and gating networks of MOME was 30:25:10 and 30:10:6, respectively, for the SRU dataset and the best structure for the expert and gating networks was 50:20:10 and 50:10:5, respectively, for the subset of the HODA dataset.

Testing MLP and MOME with the best topologies on the previously described test set, revealed that the recognition rate on the SRU dataset for MOME is 97.73%, which, for the case of single MLP, is 95.41%. Furthermore, the result of MOME is 96.48%, which has 2.31% improvement in recognition rate with respect to single MLP on the subset of the HODA dataset. In which the improvement of recognition rate is perceivable on both datasets. This is a remarkable point in the application of mixture models, as implementing a mixture of some simple MLP is much more beneficial than using a single MLP, that it, serves better the aim to study the effects of combining classifiers.

Confusion matrix can be used to realize the distribution of errors across the classes (Shipp and Kuncheva, 2002). Table 3a and b shows the confusion matrix of the recognition results for the most successful MOME network on the both datasets. For instance, two of the most misrecognized digits belong to Digits 3 and 4 (Table 3a). As shown in Table 3a, the network mistakes 7 images of Digit 3 for Digit 4, and it also mistakes 5 images of Digit 4 for Digit 3.

Scrutinizing the number of experts in MOME

The number of experts in the architecture of MOME must be specified. Hence, to make a decision about the number of experts in the network, we used different number of experts on the both datasets. The results of this experiment are presented in Table 4.

As shown in Table 4, using the best topology for experts and gating network, the best results are achieved when number of experts is 6 on the SRU and 5 on the subset of the HODA datasets, respectively.

Comparative evaluation

Finally, we would like to compare the performance of the proposed method with respect to other combination strategies in the literature of Persian handwritten digit recognition. In these experiments, because of our

Table 1a. Error rates (%) of the MLP with different number of PCA components for the SRU database.

Number of input neuron	15	20	30	40	50
Error rate	6.81	5.74	**4.76**	4.81	4.98

Error rates are the average over five times runs, each time trained with different random initial weights.

Table 1b. Error rates of the MLP with different number of PCA components for the subset of the HODA dataset.

Number of input neurons	30	40	50	60	70
Error rate (%)	7.26	6.31	**5.71**	5.73	5.85

Error rates are the average over five times runs, each time trained with different random initial weights.

Table 2a. Error rates of different topologies of MOME on the SRU dataset.

SRU dataset	Number of hidden layer neurons for gating	4	7	10	13	16	18
	Number of hidden layer neurons for experts	15	20	25	30	35	40
	Error rate (%)	3.18	2.54	**2.38**	2.61	2.59	2.91

Table 2b. Error rates of different topologies of MOME on the subset of the HODA dataset.

Subset of the HODA dataset	Number of hidden layer neurons for gating	6	8	10	12	14	16
	Number of hidden layer neurons for experts	10	15	20	25	30	35
	Error rate (%)	3.91	3.76	**3.47**	3.73	3.81	4.11

Table 3a. Confusion matrix of the best MOME network for 10 class of the handwritten digit recognition on the SRU dataset.

Class no.	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	208	0	0	2	0	5	0	0	0	0
1	0	215	0	0	0	0	0	0	0	0
2	0	2	210	3	0	0	0	0	0	0
3	0	0	2	208	<u>5</u>	0	0	0	0	0
4	0	0	2	<u>7</u>	205	1	0	0	0	0
5	2	0	0	1	1	211	0	0	0	0
6	0	0	1	1	0	2	206	1	0	4
7	0	0	0	0	0	0	0	215	0	0
8	0	0	0	0	0	0	0	0	215	0
9	0	0	0	0	0	1	0	0	2	212

proposed model is in the category of dynamic structure MOME model and base classifiers in used fusion methods. In this experiment, for the fusion methods, we use a set of 6 and 5 MLP as the base classifiers on the SRU dataset and subset of the HODA datasets, respectively. The best topologies of single MLP are assumed for the base classifiers with respect to their datasets. For diversifying base classifiers, the weights of MLP neural networks are initially set to small random values. In MOME method, the best structure for the expert and gating networks was used on both datasets. Comparison between recognition rate (%) of the MOME and proposed method in the category of selection methods and fusion combination methods implemented and tested on both datasets. The results are tabulated in Table 5. Each result is the average of ten times testing

Table 3b. Confusion matrix of the best MOME network for 10 class of the handwritten digit recognition on the subset of the HODA dataset.

Class no.	0	1	2	3	4	5	6	7	8	9
0	183	0	0	1	0	0	0	0	0	0
1	0	414	1	0	0	0	3	0	0	1
2	0	2	919	22	11	0	14	1	0	3
3	0	0	10	799	5	0	1	0	0	1
4	2	0	3	16	658	7	3	0	0	0
5	4	0	0	1	0	208	1	0	1	2
6	0	3	7	4	3	1	906	3	0	13
7	0	0	1	3	0	1	1	359	2	0
8	0	0	0	0	1	0	0	0	184	0
9	0	6	0	4	0	1	9	0	0	551

Table 4. Error rate of MOME with different number of experts on the validation set of both datasets.

Number of experts	3	4	5	6	7
Error rate (%) on the SRU dataset	3.71	3.36	3.19	**2.38**	2.62
Error rate (%) on the subset of the HODA dataset	4.53	4.11	**3.47**	3.73	3.84

Each result is the average of ten runs with different random initial weights.

Table 5. Comparison between recognition rate (%) of the MOME and other fusion methods on both datasets.

Applied method	Fusion strategy					Selection strategy
	Max	Min	Average	Product	DT	MOME
Recognition rate (%) on the SRU dataset	95.83	95.64	95.75	95.90	96.34*	97.73*
Recognition rate (%) on the subset of the HODA dataset	94.41	94.36	94.49	94.56	95.27*	96.48*

*The highest recognition rate in each of fusion and selection strategies.

the corresponding model.

As expected, our method has the highest recognition rate of 97.73% on the SRU dataset which is 1.39% higher than best method of fusion strategies, that is, DT. On the subset of the HODA dataset, the recognition rate of our method is 96.48% whereas for the case of DT method is 95.27%. As obviously showed, selection strategies are far more effective than fusion strategies. Altogether, in the case of selection strategies, because of input signal directly motivate integration outputs, it is anticipated to get more efficient recognition rate than fusion strategies.

Conclusion

This paper presented the use of combining neural network method to improving accuracy of ensemble neural networks for classification of Persian handwritten digits. Here, we aim to improve the prediction efficiency by using a modified version of MOE. Unlike the standard version of MOE, which uses the linear network as experts and gating network, our method, MOME, uses the MLP instead of linear networks. To evaluate our proposed method, we use two public domain databases to classify 10 different digit images. Taking consideration in these results on used dataset, the recognition rate of our proposed model was strongly increased with respect to single MLP. Experiments are conducted with different number of experts in MOME to achieving the desirable number of subproblem, experts becoming specialized in these subporblems. Comparison with other related fusion methods in the literature of the combining methods like Max, Min, Average, Product, Decision Templates, also demonstrated the improved performance of MOME and illustrates that it is a rich combining method.

REFERENCES

Alpaydin E, Jordan MI (1996). Local linear perceptrons for classification. EEE Trans. Neural Networks. 7(3): 788-792.

Amin A (1998). Off-line Arabic character recognition: the state of the art. Pattern Recognition. 31: 517–530.

Aradhya VNM, Kumar GH, Srihari SN (2008). Multilingual OCR system for South Indian scripts and English documents: An approach based on Fourier transform and principal component analysis. Eng. Appl. Artificial Intell., 21: 658–668.

Borji A, Hamidi M, Mahmoudi F (2008). Robust handwritten character

recognition with features inspired by Visual Ventral Stream. Neural Proc. Lett.. 28(2): 97-111.
Ebrahimi A, Kabir E (2008). A pictorial dictionary for printed Farsi sub words. Pattern Recogn. Lett., 29: 656–663.
Ebrahimpour R, Sharifizadeh F (2009). Persian handwritten digit recognition with classifier fusion: class conscious versus class indifferent approaches. World Aca. Sci. Eng. Technol., 57: 552-559.
Ghaderi R (2000). Arranging simple neural networks to solve complex classification problems. Ph.D. Thesis. Surrey University.
Glucksman HA (1967). Classification of mixed-font alphabets by characteristic loci. In: Proc. IEEE Comput. Conf., pp. 138–141.
Haykin S (1998). Neural Networks-A Comprehensive Foundation. second ed., Prentice-Hall, 842 p.
Ho TK, Hull JJ, Srihari SN (1994). Decision combination in multiple classifier systems. IEEE Trans. Pattern Anal Mach Intell., 16(1): 66–75.
Jacobs RA, Jordan .I, Nowlan SJ, Hinton GE (1991). Adaptive mixtures of local experts. Neural Comput., 3: 79-87.
Khosravi H, Kabir E (2007). Introducing a very large dataset of handwritten Farsi digits and a study on their varieties. Pattern Recogn. Lett., 28: 1133–1141.
Kittler J, Hatef M, Duin RPW, Matas J, (1998). On combining classifiers. IEEE Trans. Pattern Anal. Mach. Intell., 20(3): 226–239.
Knoll AL (1969). Expriments with 'characteristic loci' for recognition of handprinted characters. IEEE Trans. Comput., C-18: 366-372.
Kuncheva LI, Bezdek JC, Duin RPW (2001). Decision templates for multiple classifier fusion: An experimental comparison. Pattern Recogn., 34(2): 299-314.
Kuncheva LI (2004). Combining pattern classifiers: methods and algorithms. John Wiley & Sons. Inc. 350 p.
Liu CL (2005). Classifier combination based on confidence transformation. Pattern Recogn., 38: 11-28.
Liu CL, Nakashima K, Sako H, Fujisawa H (2003). Handwritten digit recognition: benchmarking of state-of-the-art techniques. Pattern Recogn., 36: 2271-2285.
Liu CL, Suen CY (2009). A new benchmark on the recognition of handwritten Bangla and Farsi numeral characters. Pattern Recogn., pp. 3287-3295.
Martinez A, Kak A (2001). PCA versus LDA. IEEE Trans. Pattern Anal Mach. Intell., 23(2): 228-233.
Ng KC, Abramson B (1992). Consensus diagnosis: a simulation study. IEEE Trans. Systems Man Cybernet., 22: 916-928.
Pan WM, Bui TD, Suen CY (2009). Isolated handwritten Farsi numerals recognition using sparse and over-complete representations. In: Proc. 10th Int. Conf. on Document Analysis and Recognition. pp. 586-590.
Shipp CA, Kuncheva LI (2002). Relationship between combination methods and measures of diversity in combining classifiers. Information Fusion. 3: 135-148.
Soltanzadeh H, Rahmati M (2004). Recognition of Persian handwritten digits using image profiles of multiple orientations. Pattern Recogn. Lett., 25: 1569-1576.
Suen CY, Izadinia S, Sadri J, Solimanpour F (2006). Farsi script recognition: a survey, in: Proc. of the Summit on Arabic and Chinese Handwriting Recognition, University of Maryland, College Park, MD. pp. 101-110.
Suen CY, Nadal C, Mai TA, Legault R, Lam L (1990). Recognition of totally unconstrained handwritten numerals based on the concept multiple experts. In: Proc. IWFHR. pp.131-143.
Tax DMJ, Breukelen MV, Duin RPW, Kittler J (2000). Combining multiple classifiers by averaging or by multiplying. Pattern Recogn., 33: 1475-1485.
Trier OD, Jain AK, Taxt T (1996). Feature extraction methods for character recognition-a survey. Pattern Recogn., 29(4): 641-662.
Woods K, Kegelmeyer WP, Bowyer K (1997). Combination of multiple classifiers using local accuracy estimates. IEEE Trans. Pattern Anal. Mach. Intell., 19: 405-410.
Xu L, Krzyzak A, Suen CY (1991). Associative switch for combining multiple classifiers. In: Proc. Int. Joint Conf. on Neural Networks. 1: 43-48.
Xu L, Krzyzak A, Suen CY (1992). Methods of combining multiple classifiers and their application to handwriting recognition. IEEE Trans. Systems Man Cybernet, 22: 418-435.