Speaker Recognition using Multi-Layer Perceptron, Time Delay Neural Network, Self-Organizing Map and Concurrent Neural Networks

Angel Cațaron
Dept. of Electronics and Computers, TRANSILVANIA University of Brasov, Romania
e-mail: cataron@vega.unitbv.ro

Stelian Mătase
Dept. of Electronics and Computers, TRANSILVANIA University of Brasov, Romania
e-mail: matase@vega.unitbv.ro

***Abstract* – Speech recognition represents the finding of a person identity based on analysis of his spoken words. In this paper we use the *Concurrent Neural Networks (CNN)* for the task of speaker recognition. This model is a *winner-takes-all* collection of individually trained neural networks. Each network of the system provide best results for one class of input partterns only. As basic components of such a system, we used, in distinct experiments, the Multi-Layer Perceptron (MLP), the Time-Delay Neural Network (TDNN) and the Kohonen Self-Organizing Map (SOM). We also performed similar tests training single neural networks as whole-task recognizers. We used a speech database called SPEECHDATA with spoken words collected from 25 persons, males and females. The words are the digits from zero to nine, plus 'nought' and 'oh'. Each word was pronounced 20 times by each speaker. The experiments proved a significant increase of the recognition scores of our CNN model by comparation to the individual neural networks.**

***Index Terms* – Concurrent Neural Networks, Kohonen Self-Organizing Map, Multi-Layer Perceptron, speaker recognition, Time-Delay Neural Network.**

I. INTRODUCTION

Previous speech and speaker recognition works [6] proved that the basic models of neural networks, such as MLP, TDNN or SOM, have acceptable performances for isolated word recognition, but quite poor for speaker recognition. The second task was proved to be more difficult, because it is necessary to distinguish between voice characteristics of different speakers.

It is obvious that reducing the number of persons, the speaker recognition task becomes easier, because the decision is less complex.

The Concurrent Neural Networks model consists in a collection of small, specialized neural networks, trained to recognize very well the patterns from the class that they were trained for and rejects the patterns which belong to any other class. The advantage of using specialized neural networks is the smaller dimension and, as experiments proved, a better recognition accuracy.

II. CONCURRENT NEURAL NETWORKS

Let us consider the set $X \subset \Re^p$ which is formed by M feature vectors from a p-dimensional Euclidian space, that is, $X=\{\mathbf{x}_1, \mathbf{x}_2 \ldots \mathbf{x}_M\}$, $\mathbf{x}_i \in \Re^p$, $1 \le i \le M$. CNN are a collection of neural networks. They are trained individually and the recognition decision is based on a *winner-takes-all* strategy.

We consider each of the feature vectors from the set X are *a priori* known to belong to one of the n classes, that is

$$X = X_1 \cup X_2 \cup X_n$$

and

$$X_1 \cap X_2 \cap X_n = \Phi,$$

where X_1, X_2 ... X_n are subsets of X and can be used as training pattern sets for each of the n neural networks (fig. 1).

The CNN global training technique is a supervised one, and for the individual networks can be used their own training algorithms.

In figure 1, n is the number of parallel performing neural networks, that is the total number of the pattern classes.

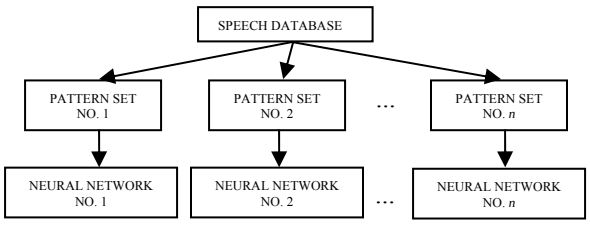

Fig. 1. The CNN model used in the training phase.

The database consists of the pre-processed speech signal, that is the set X. The pattern sets extracted from this database, that is the subsets X_1, X_2 ... X_n, are inputs of the n networks in

the training phase. The training algorithm of the CNN is the following:

Step 1. Create the database by pre-processing the voice signal.

Step 2. Extract the training pattern sets from the database. If necessary, add the desired outputs.

Step 3. Use the specific training algorithm for each individual neural network from the CNN using the training sets from step 2.

In the recall phase, each network should be activated by the patterns from its corresponding class only. The CNN performs a selection of the outputs from the individual networks (fig. 2). The selection means finding the strongest response. The selected network is declared as *winner* and its index is the class index associated to the test pattern. This classifying method suppose that the number of classes is known before the training phase and for each class is available a sufficiently number of patterns. The recall algorithm is the following:

Step 1. Create the input pattern by pre-processing the voice signal.

Step 2. Apply the pattern in parallel to the *n* trained networks.

Step 3. Find the best response network using the *winner-takes-all* selection strategy. The network index is the patterns' class index.

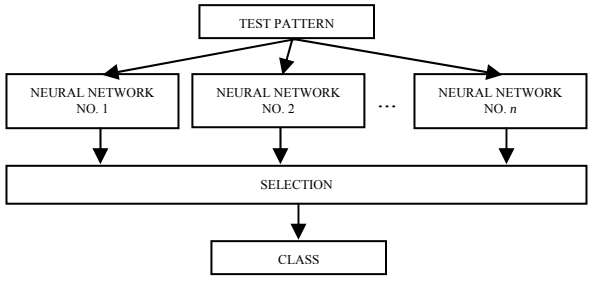

Fig. 2. The CNN recall model.

In the experiments presented in this paper we used MLP, TDNN and SOM as basic component neural networks of the CNN. We used the backpropagation of error training algorithm for MLP and TDNN. For SOM we used the Kohonen training procedure.

MLP-CNN and TDNN-CNN use pairs of input-output vectors, that is training patterns and desired responses. The *positive patterns* belong to the class having the same index as the component network of the CNN. A *negative pattern* belongs to any other class. For an efficient training, we balanced the number of positive and negative patterns. We used an equal number of positive and negative examples. The classification decision is based on the selection of the most accurate response.

SOM use an unsupervised training algorithm and therefore SOM-CNN use positive training patterns only. As previously, each component network is trained with its dedicated pattern set. The classification decision is based on the minimum quantization error. The network which generate the minimum is selected as winner.

III. THE SPEECH DATABASE

In our speaker recognition experiments we used a speech database. It consists of 20 repetitions of 12 words ("one"-"nine", "zero", "nought" and "oh") spoken by 25 talkers, giving a total size of 6000 utterances. The database, SPEECHDATA, contains data collected under controlled conditions, with a minimum amount of noise interference. The talker's speech was recorded on a proffesional cassette tape recorder using a high-quality microphone. The recorded voice signal was then digitised using a 12-bit, analog/digital converter, at a 7.5 KHz sampling rate.

The speech pattern extraction was based on the cepstral analysis [3]. The signal was first transformed using a Fast Fourier Transform (FFT), then was applied to a Mel-scale filterbank. The Mel-scale is a non-linear frequency scale reflecting the human auditory system perception capabilities and is related to the normal frequency scale using the relation

$$F_{MEL} = 2595 \log_{10}\left(1 + \frac{F_{HZ}}{700}\right) \qquad (1)$$

The speech signal was split into 20 ms frames using overlapped Hamming windows and a standard radix-2 decimation-in-time FFT algorithm was used in order to compute the short-time spectrum. The spectral output from the filterbank was transformed to cepstral domain using a discrete cosine transform (DCT). The Mel-scale filterbank outputs Y_j were computed by composing the short-time magnitude spectrum using triangular Mel-scale filterbank and the weighted filterbank components falling within each band. The Mel-frequency Cepstral Coefficients C_i were computed using the following DCT:

$$C_i = \sum_{j=1}^{N}\left(\log|Y_j|\cos\left(\frac{\pi i}{N}(j-0.5)\right)\right) \qquad (2)$$

with the condition $1 \le i \le M$, where N is the number of filters in the filterbank and M is the number of desired cepstral coefficients. For each frame P a delta coefficient d_P computed with the following relation were used:

$$d_P = \frac{\sum_{i=1}^{M} i\left(C_{P+i} - C_{P-i}\right)}{2\sum_{i=1}^{M} i^2} \qquad (3)$$

For the beginning and the end of the word, the delta coefficients were computed using simple first-order differences:

$$d_p = C_{P+1} - C_P, \quad P < M \qquad (4)$$

$$d_P = C_P - C_{P-1}, \quad P \ge N_F - M \qquad (5)$$

where N_F is the frames number in the utterance.

For SPEECHDATA, the values of N and M were 16 and 8. A frame pattern consisted of 8 cepstral coefficients, 1 coefficient representing Y_j and 9 delta coefficients. The speech signal of each word was processed in 15 overlapped Hamming windows and resulted 270 feature coefficients per word.

IV. EXPERIMENTS

The experiments described in this section present the capabilities of MLP, TDNN and SOM trained as individual speaker recognizers comparing to CNN using MLP, TDNN and SOM as basic components.

We used SPEECHDATA as training and testing database. Each of the 25 speakers pronounced the 12 words 20 times. We used 8 repetitions out of 20 for training and the remaining 12 repetition for the testing phase.

In the first set of experiments we tested MLP, TDNN and SOM as single networks in distinct experiments. The imput layer of these neural networks consisted of 270 units, fitting to the dimension of the vectors from the SPEECHDATA.

We trained MLP and TDNN using the well known supervised procedure named backpropagation of error. Therefore, we associated to each training pattern a desired response of the neural network. We designed the MLP and TDNN as having 270 input units, two hidden neuron layers and an output layer with 25 units, associating one output unit to each speaker. The words spoken by the first talker should activate the first output unit and let the other units inactive, the words spoken by the second talker should activate the second output unit and let the other units inactive, etc. In the ideal case, the response of an active output unit was 1 and the response of the inactive output was 0, but for the recognition we agreed with tolerating a difference of 0.3 form these thresholds. The desired response associated to the words from the first class was (1,0,0,...,0), the desired response associated to the words from the second class was (0,1,0,...,0), etc. Therefore, we considered active an output neuron that generated a response between 0.7 and 1 and inactive an output neuron which generated an output between 0 and 0.3. In the other cases, the response was undecided and we considered the network unable to classify the input pattern.

We trained the SOM using the unsupervised Kohonen Self-Organized Map training algorithm. After the training phase, each output unit was calibrated with a supervised method using the training patterns set. The classification decision of a test pattern was drawn by finding the best matching unit according to the minimum Euclidian distance.

The recognition scores obtained to these tests are presented in the table I in the column named "Individual network".

TABLE I
THE RECOGNITION RATES IN THE SPEAKER RECOGNITION EXPERIMENTS USING MLP, TDNN AND SOM INDIVIDUALLY AND AS BASIC COMPONENTS OF A CNN.

	Individual network	CNN
MLP	71.25%	86.22%
TDNN	57.28%	89.31%
SOM	52.86%	90.42%

In the second set of experiments, we tested the CNN for the same speaker recognition task.

The training and testing methods that we used for MLP-CNN and TDNN-CNN are similar and we will describe them together. We will present then the methods that we used for SOM-CNN.

We decided to use 25 neural networks for MLP-CNN and TDNN-CNN, one for each speaker class. Each component neural network consisted of a 270-units input layer, two hidden layers and a 25-units output layer. We also used the backpropagation of error training algorithm for these neural networks, but we created distinct training patterns sets for each of them, defining two types of input patterns. We named *positive examples* the input patterns belonging to the class with the same index as the component neural network that they were applied to. The input patterns from the other classes applied to the same neural network were named *negative examples*. According to this definition, the patterns from class 1, extracted from the words spoke by speaker 1, were positive examples for the neural network with index 1 (fig. 1) and were negative examples for the other neural networks. The desired responses of the positive examples had the same format as described in the first set of experiments. We associated the desired response (1,0,0,...,0) to the positive examples of the neural network 1 and the desired responses (0,1,1,...,1) to the negative examples of neural network 1. We followed a similar procedure for the other component networks. In order to balance the number of positive examples with the larger number of negative examples, we repeated 24 times each positive example in each training patterns set. In the recognition phase, a test pattern should activate only one neural network, that is activated neural network provides an output close to the response of a positive

example. Every other neural network from the CNN should provide outputs close to the response of their negative examples. We draw a decision only when exactly one out of the 25 neural networks provided a positive response and all the other provided negative responses. Else, the response was undecided and the pattern was not recognized.

In figure 3 we presented the average recognition rates for the 25 speakers in two typical experiments using TDNN and TDNN-CNN. It can be seen that TDNN-CNN increase the average recognition rate, though it is possible that TDNN provide better results for some individual speakers.

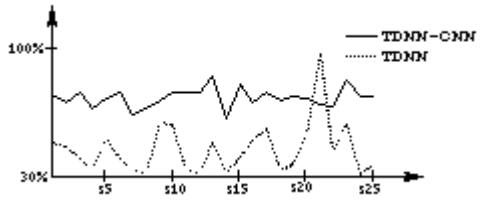

Fig. 3. The average recognition scores for the 25 speakers in two tipical recognition experiments using TDNN and TDNN-CNN.

The training of the SOM-CNN was much simpler. We created 25 training patterns sets and we used the Kohonen Self-Organising Map training algorithm independently for each of the 25 maps. For the recognition, the test pattern was applied in parallel to every SOM. The map providing the least quantization error was decided to be the winner and its index were the class index that the pattern belonged to.

In the table I we presented the recognition rates for the MLP-CNN, TDNN-CNN and SOM-CNN in the column named "CNN".

V. CONCLUSIONS

This paper presented a comparative study of the speaker recognition performances of MLP, TDNN and SOM, used individually and as basic components of CNN.

The test data consisted of a database named SPEECHDATA, obtained from the voice signal acquired in a relatively noise-free room environment. The database comprised 12 isolated words, the pronounciation of the digits "zero" to "nine" plus "nought" and "oh". Each word was spoken 20 times by 25 persons. These words were parametrized into time sequrnces of 15 frames of 18-dimension feature vectors, using the Mel-scale cepstral analysis.

The experiments showed that the recognition rates obtained with CNN are superior with more than 15% to the recognition rates of the basic neural networks models used as whole-task recognizers. MLP performed better than TDNN and SOM, but SOM-CNN recognition scores were around 4% better than MLP-CNN and more than 1% than TDNN-CNN.

These experiments proved that the use of specialized neural networks for each class leads us to better classification rates, though the global training phase takes a longer time.

ACKNOWLEDGMENT

The authors would like to thank Dr. F.J. Owens of the Universiry of Ulster at Jordanstown for providance of the speech database.

REFERENCES

[1] C.M. Bishop, *Neural networks for pattern recognition*. New York: Oxford University Press, 1995.

[2] A. Cataron and V.-E. Neagoe, "Concurrent Neural Networks for Speaker Recognition". Proceedings of IEEE International Conference on Telecommunications ICT2001, Special Sessions, Bucharest, Romania, pp. 252-257, 2001.

[3] J.R. Deller, J.G. Proakis and J.H.L. Hansen, *Discrete-time processing of speech signals*. Upper Saddle River, New Jersey: Prentice Hall, 1987.

[4] S. Haykin, *Neural networks - A comprehensive Foundation*. New York: Macmillan College Publishing Company, 1994.

[5] T. Kohonen, "The self-organizing map". *Proceedings of the IEEE* 78, pp. 1464-1480, 1990.

[6] F.J. Owens, R. Andonie, G.H. Zheng, A. Cataron and S. Manciulea, "A comparative study of the multy-layer perceptron, the multi-output layer perceptron, the time-delay neural network and the Kohonen self-organizing map in an automatic speech recognition task". *Proceedings of EIS'98 International ICSC Symposium on Engineering of Intelligent Systems*, ICSC Academic Press, Tenerife, Spain, February 11-13, pp. 624-629, 1998.

[7] L. Rabiner and B.-H. Juang, *Fundamentals of speech recognition*. Englewood Cliffs, New Jersey: Prentice Hall, 1993.

[8] A. Waibel., T. Hanazawa, G. Hinton, K. Shikano and K.J. Lang, "Phoneme recognition using time-delay neural networks". *IEEE Transactions on Acoustics, Speech, and Signal Processing* ASSP-37, pp. 328-339, 1989.

[9] J.M. Zurada, *Introduction to Artificial Neural Systems*. St. Paul, Minessotta: West Publishing Company, 1992.