Scientia Iranica, Vol. 14, No. 6, pp 571–578

SCIENTIA IRANICA

Research Note

A New Bidirectional Neural Network for Lexical Modeling and Speech Recognition Improvement

M.R. Yazdchi[1], S.A. Seyyed Salehi[2] and R. Zafarani[*]

One of the most important challenges in automatic speech recognition is the case of mismatch between training and test data. Conventional methods for improving recognition robustness seek to eliminate or reduce the mismatch, e.g. enhancement of the speech by adapting the statistical models. Training the model in different situations is another example of these methods. The success with these techniques has been moderate compared to human performance. In this paper, an inspiration from human listeners created the motivation to develop and implement a new bidirectional neural network. This network is capable of modeling the phoneme sequence, using bidirectional connections in an isolated word recognition task. This network can correct the phoneme sequence obtained from the acoustic model to what is learned in the training phase. Acoustic feature vectors are enhanced, based on the inversion techniques in neural networks, by cascading the lexical and the acoustic model. Speech enhancement by this method has a remarkable effect in eliminating mismatch between the training and test data. The efficiency of the lexical model and speech enhancement was observed by a 17.3 percent increase in the phoneme recognition correction ratio.

INTRODUCTION

A comparison of machine speech recognition with human listeners for a speaker independent task shows that machine speech recognition is acceptable for a simple task (connected digit recognition). However, word error starts to increase with a raise in task difficulty. Its difficulty can be measured in a number of ways, including perplexity (defined as number of words allowed to follow a given word), even if there are no environmental effects. Human listeners perform well under various conditions [1]. The speech recognition task becomes more difficult where there are mismatches between the training and testing conditions of the recognition systems. The performance of speech recognition systems trained in clean/controlled conditions reduces drastically during testing under noisy conditions. Previous research has revealed that human listeners are able to comprehend speech which has undergone considerable spectral excisions. For example, speech is understandable, even if it has been high or low-pass filtered [2]. These observations suggest that there is enough redundancy in the speech signal for reasonable recognition. Therefore, listeners can recognize the speech with only a fraction of spectral-temporal information present in it [3]. Conventional techniques used in these cases include; Multi-conditional training, speech enhancement and adaptation of statistical models for the speech unit. In the first method, the automatic speech recognition system is trained under different conditions. Simplicity is its main advantage. However, its main flaw is the increase in recognition error caused by various types of training data. In the speech enhancement method, speech data is enhanced in such a way as to eliminate the mismatches between training and test data. Neural networks which are able to model nonlinear functions, have shown an effective performance in speech enhancement [4]. Adaptation techniques reduce mismatches under training and test conditions by modifying the model parameters for new speakers or conditions. Observations of the human auditory system have led to a novel approach: The missing data [5]. In this method, primary recognition (coarse and holistic recognition) is accomplished by reliable regions. Unreliable regions are corrected

1. *Department of Biomedical Engineering, University of Isfahan, P.O. Box 81745, Isfahan, I.R. Iran.*
2. *Faculty of Biomedical Engineering, Amirkabir University of Technology, Tehran, I.R. Iran.*
*. *Corresponding Author, Department of Computer Engineering, University of Isfahan, P.O. Box 81745, Isfahan, I.R. Iran.*

afterwards. This action is iterated until the recognition is completed and the primary recognition is modified to the final recognition (fine recognition). This paper is an effort to design a lexical model, based on a new bidirectional neural network. This network, inspired by the parallel structure of the human brain, processes information by having up-down connections, in addition to bottom-up connections. This network tries to get closer to the performance, flexibility, correctness and reliability of the human auditory system. In this network, up-down links improve the input (phoneme sequence) in successive iterations by a new technique. Cascading this network, as a lexical model with the acoustic model, results in multiple pronunciations (alternative pronunciations) to be converted to canonical pronunciation. These pronunciations are also corrected to the hand-labeled phoneme sequences. Canonical and alternative pronunciations are, as follows:

1. **Canonical Pronunciation**: Is also known as phonemic transcription, which are the standard phoneme sequences assumed to be pronounced in read speech. Pronunciation variations, such as speaker variability, dialect or coarticulation in conversational speech, are not considered;
2. **Alternative Pronunciation**: Actual phoneme sequences pronounced in speech. Various pronunciation variations, due to the speaker or conversational speech, can be included.

In this method, there is no need for large speech databases. Neural networks are capable of making appropriate decisions for the test data not being available in the training data. This is based on what is learned in the training phase [6]. Cascading the following lexical model and acoustic model, will have remarkable results in acoustic feature vectors (acoustic data) enhancement. In this method, by using the inversion techniques in neural networks, the corrected phoneme sequence from the lexical model is used to enhance feature vectors. This enhancement re-increases the phoneme recognition correction ratio. In the following sections, some inversion approaches in neural networks are presented first. Then, speech data and feature extraction are described and the acoustic model is presented. After that, the lexical model is described and two methods of speech enhancement using neural network inversion techniques are presented. Finally, a discussion of the presented work is given.

INVERSION TECHNIQUES (ORGANIZING BIDIRECTIONAL NEURAL NETWORKS)

In this section, two techniques of neural network inversion are described, in order to recognize input from output. In the first method, the error between the desired output and the actual output is measured after the training phase. This error is back propagated through hidden layers to the input layer, in order to adjust the input [7]. During this process, weights are fixed and the input is updated. The adjustment stops when there are no errors or incomplete epochs. The second technique is based on training an inverse network. During the training phase, two feed-forward networks (direct/inverse nets) are trained. The input and output of the direct network are the output and input of the inverse one, respectively [8].

Inverting Neural Networks by Gradient Input Adjustment

In this approach, the error is back propagated to update the input. A feed-forward network, with the hidden layer, H, and input and output layers, I and O, is assumed. x_k^t is the kth element of the input vector in the tth iteration. It is initiated from x_k^0 and updated, according to the following equation, in the gradient method:

$$x_k^{t+1} = x_k^t \quad \eta \frac{\partial E}{x_k^t} \qquad \{k \in I, t = 0, 1, 2, \cdots\}. \tag{1}$$

In the above, gradient error is calculated, based on the following equation:

$$\delta_j = \begin{cases} f_j'(y_j)(y_j \quad d_j) & j \in O \\ f_j'(y_j) \sum_{m \in H,O} \delta_m w_{jm} & j \in I, H \end{cases} \tag{2}$$

where f_j is the activation function of the jth neuron, y_j is the activation of the jth neuron, d_j is the target output and w_{jm} is the weight between neurons j and m. This method, like other gradient methods, has the probability of falling into local minima.

Neural Networks Inversion Based on Training of Inverse Network

Another method that is used to compute input from output, uses two feed-forward networks in reverse structures. The training phase of these networks will converge if the training data is having a one-to-one mapping. For example, if the training data have the (many-to-one) mapping, then, the direct network will converge and the inverse network will compute a value near a point in inputs where redundancy is high [9]. These networks can be trained, based on two separated back propagation algorithms or a single bidirectional algorithm [9]. A general structure of the two networks containing a single hidden layer is shown in Figure 1. Figure 2 shows the mapping between the input space and the output space in these networks.

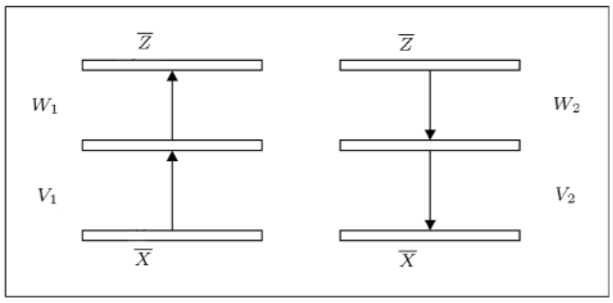

Figure 1. Two single hidden layer networks, (direct/inverse networks).

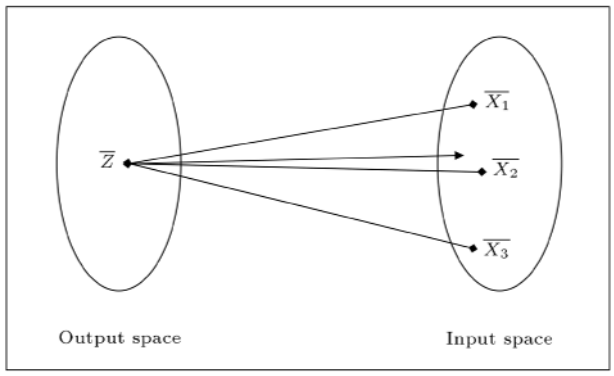

Figure 2. The mapping between the input and output space.

SPEECH DATA AND FEATURE EXTRACTION

An actual telephone database (TFarsdat) was used [10]. This database contains five sections, which are pronounced by 64 speakers. From all the pronounced words in this database, 400 words that were pronounced more than once were selected and used in the training and testing of the networks [10]. 75 percent of them were used for training and the rest were used as test data. Thus, from all words pronounced, the words of 48 speakers were used for training and the rest was used as test data. The sampling rate is 11025 Hz, which was downsampled to 8KHz (Telephone speech sampling rate). MFCC (Mel Frequency Cepstral Coefficient) features were extracted, as follows: The speech signal is segmented into 16ms-long overlapping frames with a 8ms overlapping shift [11,12].

ACOUSTIC MODEL

The acoustic model is a MLP network with two hidden layers and a neural structure of $9 \times 39 \quad 50 \quad 40 \quad 34$. In the input, 9 consecutive frames (4 consecutive previous frames, 4 consecutive frames ahead and the current (middle) frame) are presented to the network. The output is describing the middle frame using 34 neurons (number of phones). Activation functions of hidden and output layers were hyperbolic tangent functions. Their outputs were between 1 and -1, in order to make the convergence speed of the network faster. The weights were initialized, according to the Nguyen-Widrow algorithm. The network is trained, based on RPROP (Resilient Back Propagation Algorithm) [13]. The frame correction ratio of 71.34 percent was reported.

In converting frames to phonemes, the mean phoneme length, in frames, is considered important. A simple technique in the conversion is considered, since phoneme lengths are normally two frames or more. Two or more consecutive frames of one type were considered as a phoneme in the acoustic model output. Therefore, single frames were ignored. Table 1 shows the phoneme recognition ratios obtained by this technique. (Results were taken by NIST software in sequence comparison.)

LEXICAL MODEL

Until now, multi-layer feed-forward neural networks have been extensively used in pattern recognition. However, unidirectional networks lack enough capability in conditions where patterns are mixed with noise [9]. Bidirectional processes are the way in which the cortex seems to perform in making sense of the sensory input [14]. The robust pattern recognition in humans shows the capability of this method in signal clustering [9]. It seems that a holistic but coarse initial hypothesis is generated by an express forward input description and, subsequently, refined under the constraints of this hypothesis [14-16]. A bidirectional neural network for pattern completion has been applied to different applications, such as the completion of hand written numbers [17]. It was shown by this method that the appropriate training of a feed-forward neural network with up-down connections for correcting the input, can effectively rebuild the missed blocks in incomplete patterns. Therefore, a novel bidirectional neural network and its training algorithm have been designed. Missed phonemes are appropriately rebuilt in the phoneme sequence extracted from an acoustic model in an isolated word recognition task. Two major goals are settled by cascading the lexical model with

Table 1. Phoneme recognition ratios over test data from the acoustic model.

Substitution	Insertion	Deletion	Correction	Accuracy
14.7%	10.4%	13.7%	71.6%	61.1%

the acoustic model:

1. Correcting the output phoneme sequence from the acoustic model compared to the hand-labeled phoneme sequences;
2. Correcting the output phoneme sequence from the acoustic model compared to the canonical phoneme sequence.

In the first goal, cascading the lexical model with the acoustic model will improve the phoneme recognition ratios. In the second one, the multiple pronunciations are adjusted to the canonical pronunciation. In both cases, the architectures and training algorithms are completely the same. The only difference is in the training data.

Neural Network Architecture

This network is a MLP with two recurrent connections. The output and input layers have hyperbolic tangent and linear activation functions, respectively. The recurrent connection from the hidden layer to itself is for the purpose of long-term memory. The connection between the hidden layer to the input layer is for the reason of updating the input in the next iteration. Based on the partial information given, this network is capable of recalling the rest of the patterns in several iterations. (The number of iterations is dependent on the data and will be determined empirically in the training phase.) This model is used once, for adjusting the phoneme sequences in multiple pronunciation of a word to its canonical phoneme sequence. The model is also used to correct the phoneme sequences in multiple pronunciations of a word to the hand-labeled phoneme sequences.

In input, each phoneme is represented using 34 bits. This way to presentation creates the possibility of cascading this model with the lower levels (acoustic model) directly. The word boundary is determined by the hand-labeled data (isolated word task). The number of phonemes in the lengthiest word is considered as the network's input length. Silence phonemes are added to the end of the words that are shorter than this length. The output of this network is the binary presentation of the word corresponding to the input phoneme sequence. The network has a $34 \times 16 \quad 30 \quad 9$ structure. (The number of inputs show that, in the data used, the lengthiest word had 16 phonemes. The number of outputs show that a maximum of 512 words can be classified.)

Figure 3 shows the network architecture. Parts of the information are given as inputs to the network. The output is calculated in the nth iteration and the input is adjusted in the $n+1$th iteration, based on the following equations. This action is iterated N_0 times, until the rest of the pattern is revealed. N_0 (number of iterations) is dependent on the input data, determined empirically and used in the training phase. The network output is the average of all outputs during the iterations.

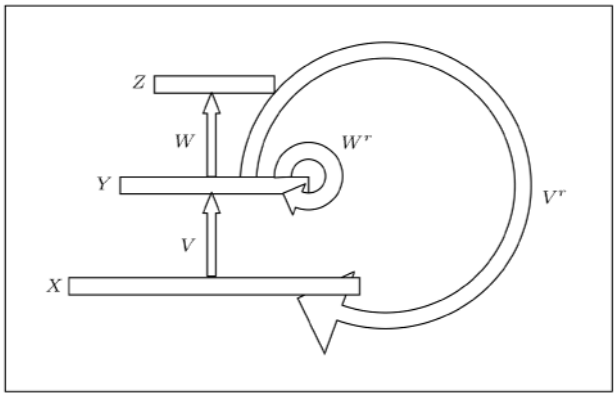

Figure 3. Neural network structure in lexical model.

$$y(j,n) = f(\sum_{i=0}^{n_I} x(i,n)v_{ij} + \sum_{k=1}^{n_H} y(k,n \quad 1)w_{kj}^r),$$

$$j = 1, \cdots, n_H,$$

$$z(k,n) = f(\sum_{j=1}^{n_H} y(j,n)w_{jk}) \qquad k = 1, \cdots, n_0,$$

$$x(i,n+1) = (1 \quad \gamma)x(i,n) + \gamma f(\sum_{j=1}^{n_H} y(j,n)v_{ji}^r),$$

$$i = 1, \cdots, n_I, \tag{3}$$

where $n = 1, \cdots, N_0$ and $0 \leq \gamma \leq 1$.

In the above equations, γ will specify the completing speed of the incomplete patterns. This coefficient is determined empirically in test and training phases. Selecting a nonproper value for it can lead to a divergence in the algorithm.

Network Training Algorithm

After the initialization of the network parameters using the following equation, the network is trained by an algorithm, which is derived from gradient and back propagation methods:

$$\hat{z}(k,n) = \frac{1}{n}\sum_{m=1}^{n} z(k,m), \qquad k = 1, \cdots, n_0,$$

$$\dot{z}(k,n) = (\hat{z}(k,n) \quad d_z(k))f'(z(k,n)),$$

$$k = 1, \cdots, n_0,$$

$$w_{jk} = w_{jk} + \eta\dot{(}z)(k,n)y(j,n),$$

$$k = 1, \cdots, n_0, \qquad j = 1, \cdots, n_H,$$

$$\dot{y}(j,n) = f'(Y(j,n))(\sum_{k=1}^{n_I} \dot{x}(k,n+1)v_{jk}^r + \sum_{k=1}^{n_0} \dot{z}(k,n)w_{jk} + \sum_{i=1}^{n_H} \dot{y}(i,n+1)w_{ij}^r),$$

$$j = 1, \cdots, n_H,$$

$$w_{ij}^r = w_{ij}^r + \eta\dot{y}(j,n)y(i,n \quad 1),$$

$$i = 1, \cdots, n_H, \qquad j = 1, \cdots, n_H,$$

$$v_{ij} = v_{ij} + \eta\dot{y}(j,n)x(i,n \quad 1),$$

$$i = 1, \cdots, n_I, \qquad j = 1, \cdots, n_H,$$

$$\dot{x}(i,n) = (1 \quad \gamma)\dot{x}(i,n+1) + \gamma(\sum_{j=1}^{n_H} \dot{y}(j,n)v_{ij})f'(x(i,n+1)),$$

$$i = 1, \cdots, n_I,$$

$$v_{ij}^r = v_{ij}^r + \eta\dot{x}(k,n)y(j,n),$$

$$k = 1, \cdots, n_I, \qquad j = 1, \cdots, n_H, \qquad (4)$$

where:

$$n = N_0 \quad 1, \cdots, 1.$$

Equation 5 shows the error measurement in the last iteration. Equation 4 is used to measure errors in the previous iterations, recursively.

$$\hat{x}(k,N_0) = \frac{1}{N}\sum_{m=1}^{N_0} x(k,m),$$

$$k = 1, \cdots, n_I,$$

$$\dot{x}(k,N_0) = (\hat{x}(k,N_0) \quad d_x(k))f'(X(k,N_0)),$$

$$k = 1, \cdots, n_I. \qquad (5)$$

Words pronounced by 48 speakers were used in the network training. In the test phase, the obtained phoneme sequences from the acoustic model were used. According to how phonemes are extracted from frames, a mean value from the frames in one phoneme is calculated and the phoneme sequence is extracted. Table 2 shows the results. It must be mentioned that the results in Tables 2 and 3 are obtained based on the phonemic (canonical pronunciation) and the hand-labeled transcription, as the desirable output, respectively. In order to compare bidirectional neural networks' abilities, the results were compared to unidirectional neural networks (Autoassociative MLP with a $34 \times 16 \quad 100 \quad 30 \quad 100 \quad 34 \times 16$ structure) and Elman Recurrent Network [18] (with a $34+34 \quad 20 \quad 34$ structure) in Tables 2 and 3 [19].

SPEECH ENHANCEMENT

Speech enhancement is motivated by the need to improve the performance of speech recognition systems in noisy conditions. Speech enhancement is a method for improving the performance of an ASR where there is mismatch in training and test data. The most common methods used for speech enhancement include spectral subtraction, HMM based methods and Kalman filtering. Spectral subtraction is the easiest technique to implement, although the method suffers from various

Table 2. Phoneme recognition ratios over test data. The hand-labeled phoneme sequence is the target output.

Model	Accuracy	Correction	Deletion	Insertion	Substitution
Unidirectional Network	67 %	74.5 %	12 %	7.5 %	13.5 %
Bidirectional Network	82.5 %	88.5 %	4.7 %	6 %	6.81 %
Elman Network	67.8 %	75.2 %	11.8 %	7.4 %	13 %

Table 3. Phoneme recognition ratios over test data. The canonical pronunciation phoneme sequence is the target output.

Model	Accuracy	Correction	Deletion	Insertion	Substitution
Acoustic Model	50.2 %	65 %	16.6 %	14.8 %	18.4 %
Unidirectional Network	57.8 %	69.2 %	14.6 %	11.4 %	16.2 %
Bidirectional Network	71.5 %	80.4 %	9.3 %	8.9 %	10.3 %
Elman Network	58.2 %	69.5 %	14.5 %	11.3 %	16 %

complexities. First of all, it requires prior assumptions about the noise estimation. A good noise estimate is not an attainable prerequisite for non-stationary noise. Another serious problem arises because of the overestimation of the noise. The problem with model and filter based techniques is their complex structure. Neural nets provide alternatives to traditional speech enhancement techniques, both in the time and frequency domain. The advantage of using neural nets for spectral domain enhancement is, in many cases, the elimination of musical noise. Neural nets are also considered suitable when there is a nonlinear mixing of noise and speech. Speech enhancement techniques with neural networks can be categorized into time-domain filtering, transform-domain mapping, state-dependent model switching and on-line iterative approaches [20]. The lexical model described in the previous section is cascaded with the acoustic model here. This corrects the phoneme sequence obtained from the acoustic model. Corrected phoneme sequence is used to enhance features, based on the inverting techniques discussed previously.

Speech Enhancement Using the Gradient Inverting Method

In this method, the acoustic model is cascaded with the lexical model described in the previous section. The lexical model is cascaded to correct the phoneme sequence to the hand-labeled transcription of them. The error between the phoneme sequence in the lexical model output and the acoustic model output is measured. This error is back-propagated to update the features and all training data are adjusted. A new acoustic model is trained over these enhanced features. The lexical model is cascaded with this new acoustic model. The results revealed that the recognition ratios are re-improved. The results are available in Table 4 and Figure 4 illustrates this method.

Speech Enhancement Using the Inversion Method of Training Inverse Networks

There is a need for the inverse network of an acoustic model in this method. Both the acoustic model and its inverse are trained in the training phase, as illustrated in Figure 5. The inverse model is capable of generating the input (features) from the output (phoneme sequence). As described before, the lexical

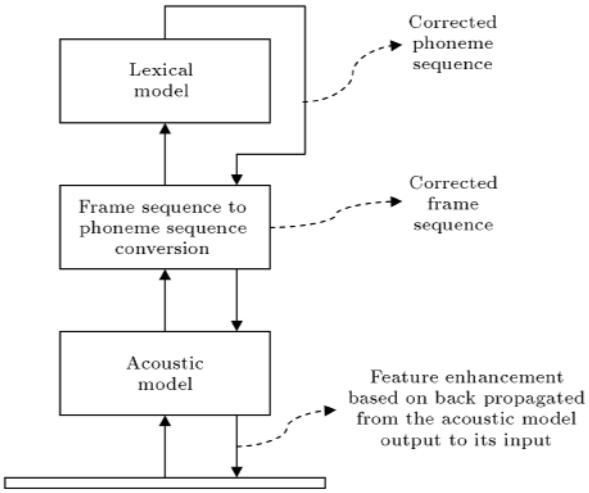

Figure 4. Cascading the lexical and acoustic model to enhance features in the gradient inversion method.

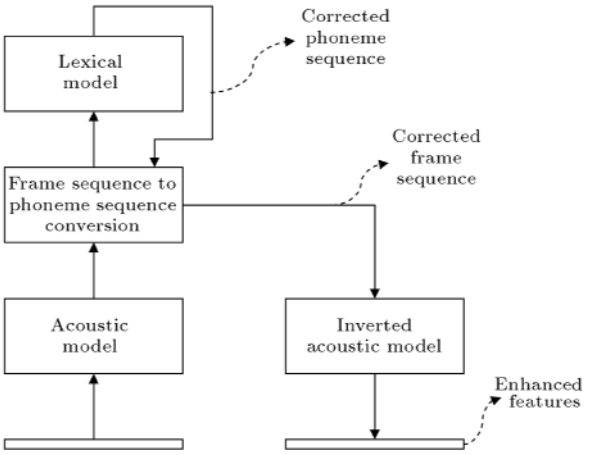

Figure 5. Cascading the lexical and acoustic model to enhance features in the inverse network training method.

model corrects the phoneme sequence of the acoustic model and features are adjusted based on this phoneme sequence. Enhanced features are obtained from the adjusted frame sequence, using the acoustic model's inverse and a new acoustic model is trained over these enhanced features. The lexical model is cascaded with this new acoustic model. An outstanding rise in phoneme recognition ratios is observable in Table 4.

Phoneme Boundary Detection

The information about phoneme boundary is completely necessary in order to adjust the frame sequence

Table 4. Phoneme recognition ratios over test data (phoneme boundary determined from hand-labeled data).

Method	Accuracy	Correctness	Deletion	Insertion	Substitution
Gradient Based	83.4 %	89.1 %	4.3 %	5.7 %	6.6 %
Inverse Network	83.9 %	89.6 %	4.1 %	5.7 %	6.3 %

Table 5. Phoneme recognition ratios over test data (phoneme boundary detection using boundary detector network).

Method	Accuracy	Correction	Deletion	Insertion	Substitution
Gradient Based	82.8 %	88.7 %	4.5 %	5.9 %	6.8 %
Inverse Network	83.1 %	88.9 %	4.7 %	5.8 %	6.4 %

from the corrected phoneme sequence. In the above, the phoneme boundary is determined by the hand-labeled transcription. A more practical way is to detect phoneme boundary by features.

A neuron (boundary detector neuron) with the sigmoid activation function, is placed in the acoustic model output layer. If the phoneme boundary is exactly in the middle of the input, this neuron will be set to '1'. If the phoneme boundary is one frame ahead or behind, this neuron will be set to '.75'. If the boundary is located 2 frames ahead or behind, the output will be set to ".25" and, otherwise, to '0.0'. Table 5 shows the phoneme recognition ratios of the previous method, using the boundary detection network.

CONCLUSION AND FUTURE WORK

Studying human perception and recognition systems shows that they have a hierarchical and bidirectional structure [14-16,21,22]. High level information has improved the flexibility and performance of image recognition systems [14].

The bidirecting of feed-forward neural networks has created remarkable improvement in their performance and the creation of dynamic basins of attraction [23-25]. In this paper, the performance of a new bidirectional neural network in an ASR was considered. Comparing the feed-forward networks' results shows the advantages of bidirectional networks. This bidirectional network is an auto/hetero associative memory capable of completing input patterns.

It has been shown that a combination of lexical knowledge and the features improves the recognition ratios. Bidirectional links create the possibility of eliminating the mismatches in input patterns. In other words, high level knowledge is used in describing low level input. The bidirectional links can be established using one of the two neural network inverting techniques. The inverse network method shows better results compared to the gradient based method. This is due to the local minimum problem. Phoneme boundary detection error in phoneme boundary detection networks results in a decrease in the recognition ratios.

The described method is used in an isolated word task. Recurrent neural networks can model time series, therefore, these networks can be applied in continuous speech recognition.

In continuous speech recognition, by adding a context layer to the network's input, the introduced bidirectional neural network will be able to model the phoneme sequence. On the other hand, by detecting word and phoneme boundaries, the given techniques can also be applied to continuous speech. In this way, all the given methods are applied after the processes of phoneme and word segmentation.

REFERENCES

1. Lippmann, R. "Speech recognition by machines and humans", *Speech Communication*, **22**(1), pp 1-15 (1997).
2. Fletcher, H., *Speech and Hearing in Communication*, New York, Van Nosstrand (1953).
3. Gong, Y. "Speech recognition in noisy environments: A survey", *Speech Communication*, **16**(3), pp 261-291 (1995).
4. Lockwood, P. and Boudy, J. "Experiments with a nonlinear spectral subtractor (NSS), hidden Markov models and the projection, for robust speech recognition in cars", *Speech Communication*, **11**(2-3), pp 215-228 (1992).
5. Parveen, S. and Green, P. "Speech recognition with missing data techniques using recurrent neural networks", *Advances in Neural Information Processing Systems*, **14**, pp 0-262 (2001).
6. Baldi, P. and Hornik, K. "Neural networks and principal component analysis: Learning from examples without local minima", *Neural Networks*, **2**(1), pp 53-58 (1989).
7. Jensen, C., Reed, R., Marks, R., EI-Sharkawi, M., Jung, J., Miyamoto, R., Anderson, G. and Eggen, C. "Inversion of feedforward neural networks: Algorithms and applications", *Proceedings of the IEEE*, **87**(9), pp 1536-1549 (1999).
8. Williams, R. "Inverting a connectionist network mapping by backpropagation of error", *Proceedings 8th Annual Conference of the Cognitive Science Society*, Lawrence-Erlbaum, pp 859-865 (1986).
9. Salehi, S.A.S. "Improving performance in pattern recognition of direct neural networks using bidirecting methods", *Technical Report*, Faculty of Biomedical Engineering, Amirkabir University of Technology, Tehran, Iran (2004).
10. Bijankhan, M. and Sheikhzadegan, M. "FARSDAT-The speech database of Farsi spoken language", *Proceeding of Speech Science and Technology Conference*, pp 826-831 (1994).

11. Rahimi Nejhad, M. "Extending and enhancing feature extraction methods in speech recognition systems", *Master's Thesis*, Faculty of Biomedical Engineering, Amirkabir University of Technology (2002).

12. Vali, M. and Seyyed Salehi, S.A. "A review over MFCC and LHCB in robust recognition of direct and telephone speech varieties", *Proceedings of the 10th Annual Conference of Computer Society of Iran*, pp 305-312 (2003).

13. Nguyen, D. and Widrow, B. "Neural networks for self-learning control systems", *Control Systems Magazine, IEEE*, **10**(3), pp 18-23 (1990).

14. Korner, E., Gewaltig, M., Korner, U., Richter, A. and Rodemann, T. "A model of computation in neocortical architecture", *Neural Networks*, **12**(7-8), pp 989-1005 (1999).

15. Koerner, E., Tsujino, H. and Masutani, T. "A cortical-type modular neural network for hypothetical reasoning-II. The role of cortico-cortical loop", *Neural Networks*, **10**(5), pp 791-814 (1997).

16. Korner, E. and Matsumoto, G. "Cortical architecture and self-referential control for brain-like computation", *Engineering in Medicine and Biology Magazine, IEEE*, **21**(5), pp 121-133 (2002).

17. Seung, H. "Learning continuous attractors in recurrent networks", *Proceedings of the 10th Annual Conference on Advances in Neural Information Processing Systems 1997*, pp 654-660 (1998).

18. Elman, J. "Finding structure in time", *Cognitive Science*, **14**(2), pp 179-211 (1990).

19. Yazdchi, M., Seyyed Salehi, S.A. and Almas Ganj, F. "Improvement of speech recognition using combination of lexical knowledge with feature parameters", *Journal of Iranian Computer Society*, **3**(3(a)) (Fall 2005).

20. Wan, E. and Nelson, A. "Networks for speech enhancement", Handbook of *Neural Networks for Speech Processing*, pp 541-541 (1998).

21. Ghosn, J. and Bengio, Y. "Bias learning, knowledge sharing", *Neural Networks, IEEE Transactions on*, **14**(4), pp 748-765 (2003).

22. Mesulam, M., *From Sensation to Cognition*, Brain, **121**(6), pp 1013-1052 (1998).

23. Saul, L.K. and Jordan, M.I., *Attractor Dynamics in Feedforward Neural Networks, Neural Computation*, **12**(6), pp 1313-1335 (2000).

24. Trappenberg, T. "Continuous attractor neural networks", *Recent Developments in Biologically Inspired Computing*, Hershey, PA: Idee Group (2003).

25. Wu, Y. and Pados, D. "A feedforward bidirectional associative memory", *IEEE Transactions on Neural Networks*, **11**(4), pp 859-866 (2000).