Proceedings of the 5th WSEAS International Conference on Signal Processing, Istanbul, Turkey, May 27-29, 2006 (pp99-104)

Arabic Phoneme Recognition Using Neural Networks

MANAL EL-OBAID[1]
Faculty of Computer Science &Eng.,
Ajman University of Science & Technology
Ajman – UAE

AMER AL- NASSIRI[2]
Faculty of Computer Science & Eng.,
Ajman University of Science & Technology
Ajman – UAE

IMAN ABUEL MAALY[2]
Faculty of Engineering,
University of Khartoum,Khartoum, SUDAN,

Abstract: - The main theme of this paper is the recognition of isolated Arabic speech phonemes using artificial neural networks, as most of the researches on speech recognition (SR) are based on Hidden Markov Models (HMM). The technique in this paper can be divided into three major steps: firstly the preprocessing in which the original speech is transformed into digital form. Two methods for preprocessing have been applied, FIR filter and Normalization. Secondly, the global features of the Arabic speech phoneme are then extracted using Cepstral coefficients, with frame size of 512 samples, 170 overlapping, and hamming window. Finally, recognition of Arabic speech phoneme using supervised learning method and Multi Layer Perceptron Neural Network MLP, based on Feed Forward Backprobagation. The proposed system achieved a recognition rate within 96.3% for most of the 34 phonemes. The database used in this paper is KAPD (King AbdulAziz Phonetics Database), and the algorithms were written in MATLAB.

Keyword: speech recognition, Arabic phonemes, Neural Networks, Signal processing, Feature extraction.

1 Introduction

Although Arabic is currently one of the widely spoken languages in the world, which is the sixth most spoken language with an estimated number of 250 million speakers [1]. Despite this fact there has been little research on Arabic speech recognition compared to other languages of similar importance. The range of the possible applications in SR is wide and different approaches in speech recognition have been adopted. They can be divided mainly into two trends: Hidden Markov model (HMM) and Neural Networks (NN). HMMs have been the most popular and most commonly used approaches [2] while NN haven't been used for SR until recently. A combination of Back propagation (B.P) learning process with feedforward neural network and supervised learning method were implemented. In the B.P. the difference between the actual response and the desired response of the output layer is propagated back by adjusting the weight structure of the entire network, so that next time a similar input token that presented the output layer response will be closer to the correct response. This is known as the Supervised Learning [3][4][5]. Section 2 shows the preprocessing and feature extraction using linear predictive coding. Section 3 describes the NN training. Section 4 discusses the networks testing and experimental results. Finally, the conclusion and future work were shown in sections 5 & 6.

2 Signal Processing for Arabic phonemes

In this part the speech data collected and the method used to transform raw speech samples into usable format is discussed.

2.1 Database Collection

Our speech data was (KAPD) Arabic Phonetics Database which created by King Abdulaziz City for Science and Technology (KACST)[6]. To study language sounds in terms of acoustics and articulator, we need to put them in a carrier, in (KAPD) the same carrier for all sounds were applied, the target sound is surrounded by a vowel where the initial and final sound is /z/. So, the token is in this form CV-VC; where C is /z/ and V is /a/, /i/ or /u/. When the target sound is initial the form is -VC, and when it is final, the form is CV-[6][7]. Modern Standard Arabic

phonemes, which are 34 phonemes including 6 vowels and 28 consonants are used in this paper. The wave files that we used from KAPD have a high quality recording at 32 KHz sampling rate and consist of 9 speakers. In this paper 4 speakers have been selected with all sounds that recorded by them. Among these files we concentrated on the files that the target sound appears in the beginning or end of the carrier to minimize segmentation errors. Figure 1 shows an example for letter "w و" in the end of the carrier.

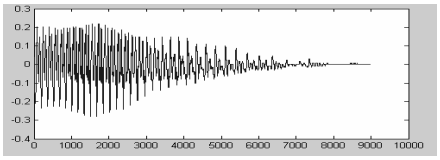

Fig.1 Waveform of letter "w و".

2.2 Raw speech segmentation

To process analog signals it is first necessary to convert them into digital form which is called analog-to-digital (A/D) conversion [8][9]. For the purpose of recognition, speech needs to be segmented into phonetic units. The segmentation of speech was depending on two methods, manual and automatic segmentation, in this paper we used manual segmentation to reduce the effect of error produced by automatic segmentation. Table 1 lists the samples range of speaker1.

Table 1 Samples range of speaker1

Sample Range	Sound	Sound code
8000,13000	الف	’
7000,12000	باء	b
7000,12000	ت	t
2000,7000	ث	th
1800,7000	ج	j

It is noticeable that all sounds do not start from sample 1, although most of them are posted in the beginning of the carrier. The reason is that there is silence in the beginning of each sound. To continue the analysis the silence must be removed, however removing this silence must not affect the original sound as shown in figures 2, 3& 4.

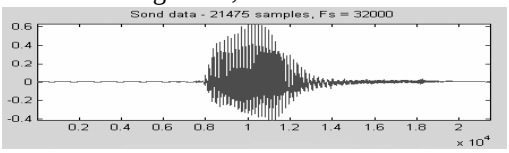

Fig.2. Full plotting of the sound *ṭ -az* with the carrier

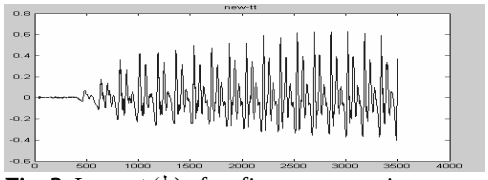

Fig..3. Letter ṭ (ط) after first segmentation (removing the carrier)

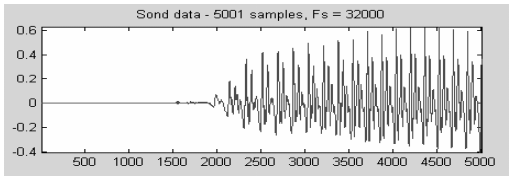

Fig.4. Letter ṭ (ط) after the second segmentation (removing the silence)

2.3 Digital signal processing

Digital signal processing is used to transform the raw signal (speech waveform) into other suitable formats for features extraction.

2.3.1 Preemphasis

The digitized speech waveform suffers from additive noise. An example of such a waveform is shown in figure 5.

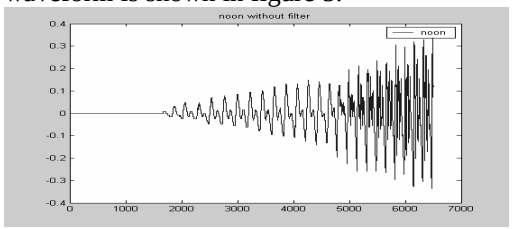

Fig.5 Letter n (ن) before filtering

In order to reduce this noise pre-emphasis is applied. Two methods are investigated, the first method is applying the FIR filter, in which we imply the application of a high pass filter [9][10], which is usually a first-order FIR of the form:

$$H(z) = 1 - az^{-1}, 0.9 \le a \le 1.0$$

The FIR has the effect of spectral flattening [12]. The selected value for *a* in our work was 0.9375. Figure 6 shows waveform of the letter n (ن) after applying preemphasis using the FIR filter.

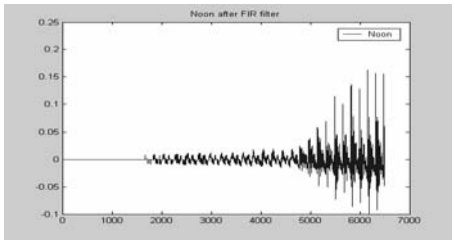

Fig.6. Waveform of the letter n (ن) after preemphasis using the FIR filter.

The second method used for preemphasis consists of three stages: The first stage is known as the Average Subtraction, which

calculates the difference between the original signal samples and the signal average as shown below:[7][10]

$$SignalAverage = \frac{1}{n}\sum_{j=1}^{n} x(i)$$

$$y(i) = x(i) - SignalAverage$$

Where n, x, and y are the number of samples, the original signal and the resulting signal after the average subtraction, respectively. The signal is then passed through the second stage, the Normalization, which divides each sample of the signal by the root of the summation of the squared samples of the signal as follows:

$$z(i) = \frac{x(i)}{\sqrt{\frac{1}{n}\sum_{i=1}^{n} x^2(i)}}$$

The purpose of the normalization is to keep signal amplitudes within acceptable range. Finally we have to calculate the signal power, in which each sample of speech is raised to the second power [10][13] as shown below:

$$w(i) = z^2(i) \qquad \text{Where} \quad 0 \le i \le n$$

Figure 7. Shows the stages of the second method.

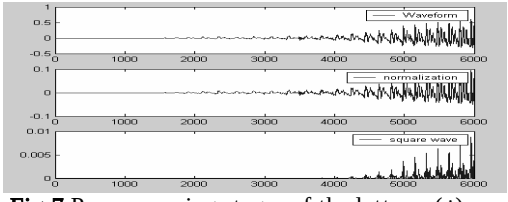

Fig.7 Pre-processing stages of the letter ن (ṉ)

The results obtained from the second method were not good when using LPC. So we have decided to continue the analysis with the results obtained from the first method.

2.3.2 Partitioning the Signal into

Frames In this step the preemphasized signal is blocked into frames of N samples, with adjacent frames being separated by M (M N) samples. The first frame consists of the first N speech samples. The second frame begins M samples after the first frame, and overlaps it by N - M samples, this process continues until all the speech is accounted for within one or more frames [9][12]. According to this, the signal, sampled at 32 kHz, is decomposed into a sequence of overlapping frames. Each frame corresponds to 16ms (512samples) of signal with a frame-overlap of 5ms (170 samples).

2.3.3 Windowing

The next step in the processing is to window each individual frame so as to minimize the signal discontinuities at the beginning and end of each frame. If we define the window as $w(n), 0 \le n \le N-1,$ where N is the number of samples in each frame, then the result of windowing is the signal

$$y_l(n) = x_l(n)w(n), \quad 0 \le n \le N-1$$

Typically the *Hamming* window is used, which has the form [9] [12] [14]:

$$w(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right), \quad 0 \le n \le N-1$$

2.3.4 LPC analysis

The feature measurements of speech signals are extracted using one of the following spectral analysis techniques: filter bank analyzer, LPC analysis or discrete Fourier Transform analysis. Since LPC is one of the most powerful speech analysis techniques [15][16] for extracting good quality features, we selected it, as we mentioned previously, to extract the features of the speech signal [9][12][11]. LPC was implemented using the autocorrelation method, in which each frame of windowed signal is auto-correlated, where the highest autocorrelation value p is the order of the LPC analysis. Typically values of p from 8 to 16 have been used [9], with $p=8$ being the value used in this paper.

2.3.5 LPC conversion to Cepstral coefficients

The features used in this system are the weighted LPC based cepstral coefficients, since they have been shown to be a more robust, reliable feature set for speech recognition than the LPC coefficients. Generally, a cepstral representation with $Q > p$ coefficients is used where Q ≈ (3/2) p. [9] in this paper p = 8 so Q = 12.

2.4 Vector quantization

Optimization of the system is achieved by using vector quantization (code book) in order to compress and reduce the variability in the feature vectors derived from the frames[9][14]. A codebook of 32-element vector was chosen, and generated by Lloyd's K-means algorithm applied on the speech samples. Table 2 shows the Quantization Error values of some selected letters. The output of this stage is the final feature used throughout.

Table 2. Quantization Error

Phoneme	Quantization Err.
x	0.00070560
d	0.00097802
s	0.00052451
f	0.00029982
m	0.0011000

3 Proposed NN for Arabic Phonemes

There are mainly two general types of functions, prediction and classification networks. In this paper, we have used classification networks, since Predictive networks known to be not very effective for speech recognition [3]. In classification network, input vectors are mapped into one of N classes. A neural network can represent these classes by N output units, of which the one corresponding to the input vector's class has a "1" activation while all other outputs have a "0" activation.

3.1 Networks architecture

In this part we are going to discuss the common features used in our networks.

- Network architecture:
  - 32 input coefficients (the quantized data), accordingly 32 neurons in the first input layer are required.
  - 34 phoneme outputs, accordingly 34 neurons in the last output layer are required
  - All activations = [-1, 1]
- Training:
  - Frames presented in serial order,
  - Learning rate schedule = optimized via search
  - Momentum used
  - Error criterion = Sum Square Error (SSE) and Mean Square Error (MSE).
- Testing:
  - test data = 34 word (male).

3.2 Transfer Functions

Nonlinear transfer functions are used since we are working with a highly nonlinear input data. Nonlinear function squash any input into a fixed range, so considered to be much more powerful, and they are used almost exclusively [3]. In this paper the sigmoid transfer function which is commonly used in Back-Propagation network were selected. The sigmoid function takes the input, that may have any value between plus and minus infinity, and squashes the output into the range 0 to 1 [3][17].

3.3 Training Procedures

Seven experiments have been implemented with BP, one will be explained in this section and the rest will be included in the results table. In experiment 6 shown in table 3, for instance, the feed forward back-propagation network (32, 100, 40, and 34) is a two-layer network in which the hidden (first) layer has 100 neurons, and the second hidden layer has 40 neurons, both hidden layers are feed forward layers. The inputs to the hidden layer are the quantized speech data which contains 32 elements for each letter in the speech and represent the input data. Figure 8 shows the B.P. network used in experiment 6, the number of neurons for hidden layers, and input/output layers.

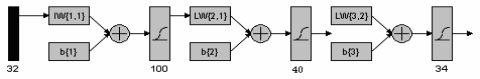

Fig.8 BP architecture of experiment 6

The training and classification of the extracted features can be implemented in several ways, the thirty four letters 32-element input vectors are defined as a matrix of input vectors. The target vector is a 34-element vector with a 1 in the position of the letter it represents, and 0's everywhere else. The network receives the 32 values as a 32-element input vector. It is then required to identify the letter by responding with a 34-element output vector each represent a letter. If recognition task success, the network should respond with a 1 in the position of the letter being presented to the network. All other values in the output vector should be 0. The log-sigmoid transfer function was picked because its output range (0 to 1) is perfect for learning to output Boolean values. Most of the training is done using both adaptive learning rate and momentum. In figure 9, the curved line on the graph shows the reliability for the network training and reaching the desired output.

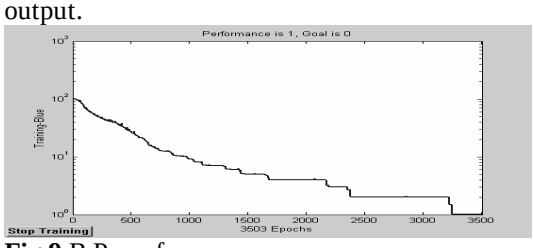

Fig.9 B.P. performance curve

4 Experimental Results

The Experiments described in this paper were fully implemented in MATLAB; several test inputs are used. The obtained results are summarized in Table 3. Testing the neural network is done by calculating the MSE

(mean square error) from the difference between the actual output values and the require target values. In the simulation process to find out which letter is the spoken one, each of the 34 letters has a particular output unit contains a positive activation. Each output unit in turn is assumed to contain a positive activation, while every other output unit is assumed to contain a negative activation. The output with the smallest MSE is chosen to be the digit spoken [11] [17].
Best result obtained with [32 100 40 34] network, with sigmoid function, tringdx training function and *mse* performance function. Traingda and Traingdx methods were used to improve the recognition rate.

Table 3. Performance, Error and Number of Layers of BP Networks

Exp.	Err. Goal	Train. Func.	neurons per layer		Training Data		Test Data	
			Hid.1	Hid.2	Perf(%)	Error	Perf.%	Er. SSE
1	0	tringdx	70	-	22	78	19	81
2	0	tringda	70	40	90.7	9.341	76	24.07
3	0	tringdx	100	34	93.8	6.224	79.6	20.4
4	0	tringdx	70	40	95	5.278	78	22
5	0	tringdx	100	34	95.8	4.281	83.3	14.703
6	0	tringdx	100	40	96.3	3.702	84.4	13.604
7	1	tringdx	120	50	93.6	6.412	79.5	20.5

Table 4 shows the simulation of the best recognition result obtained experimentally. The results in table 4 shows that the *low vowel* a "alif أ" is recognized correctly since the value in the column 1 is 0.9996 that means near to 1 which is the recognition goal. Second column, the highest value is the second value that means the *voiced bilabial stob* b "ب" is recognized correctly. But it is clear that column 4 has no good value which indicate the problem of recognizing the sound "ذ"

Table 4. selected phoneme result.

S.	A أ	b ب	to د ث	dh ذ	r ر
1	0.9996	0.0000	…….	0.0003	0.0014
2	0.0000	0.9964	…….	0.0000	0.0010
3	0.0000	0.0000	…….	0.0000	0.0000
4	0.0006	0.0004	…….	0.0000	0.0001
5	0.0000	0.0000	…….	0.0000	0.0002
6	0.0004	0.0019	…….	0.0001	0.0000
7	0.0037	0.0000	…….	0.0003	0.0001
8	0.0004	0.0005	…….	0.0001	0.0007
9	0.0000	0.0000	…….	0.0000	0.0000
10	0.0006	0.0000	…….	0.0000	0.9976

Table 5 shows the results obtained in English alphabet recognition compared with the result obtained by our system.

Table 5. Previous results of English Alphabet recognition

Owner	Perf. %	Technology
Soong [18]	92.3	NN
Dai[19]	89.3	HMM
Woodland[20]	99.3	NN
Burr [21]	85	NN
proposed BP	**96.3**	**NN**

Finally, a comparison chart between the existing method (owner's) and the suggested recognition system is shown in figure 11.

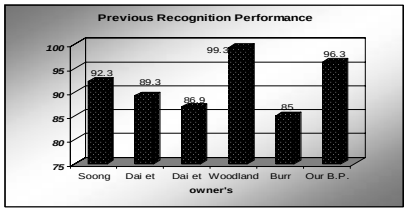

Figure 11. Previous work in English Alphabet Recognition Compaired with our BP.

5. Conclusions

This Arabic Speech recognition System is a phoneme recognition system that based on Neural Networks, programming tool was MATLAB. We have presented the Arabic Speech Recognition system based on Backpropagation MLP Neural Networks. In accordance with the results obtained, the best performance on the training data was 96.3% which obtained using BP network (32, 100, 40, and 34), with error value of 0.0011. The best recognition performance achieved on the test data is 84.4. The network architecture that attains these performances has input zone size of 32-element vector, and output size of 34 neurons. The hidden layers number and neurons depends on the network used. According to the obtained results, Training

the network with adaptive learning rate and momentum reduces the time that network needs to reach the same error with standard learning rate. The proposed Arabic speech phoneme recognition system using neural network shows promising future in this field.

6. Future Work

The results obtained indicate useful future research directions for Arabic Speech Recognition. The following points could be useful for future work

1- A larger Standard Arabic Database is needed.

2- Improving the system to a real time system is required.

3- To save time and reduce errors in future work, a good segmentation algorithm to segment phonemes from its carriers is fundamental.

4- A future work to improve the system to a speaker independent system is recommended. This depends on KAPD enlargement.

References

[1] *Katrin Kirchhoff_Jeff Bilmes_,*"Novel Approaches To Arabic Speech Recognition" pp. 2-4, report from the 2002 johns-hopkins summer workshop.

[2] F. Freitag, E. Monte, "Acoustic-Phonetic Decoding Based On Elman Predictive Neural Networks", Universitat Politècnica de Catalunya, P1.

[3] Joe Tebelskis, "Speech Recognition using Neural Networks", PhD thesis, CMU, Pittsburgh, Pennsylvania, 1995.

[4] H. Hermansky, "perceptual linear predictive (PLP) analysis of speech", Journal Acouc. Soc. Am. N. 87 (4), 1990, pp.1738-1752.

[5] B.Boudraa, and S.A. Selouani, "matrices phonétiques et, matrices phonologiques arabes", proceedings XXèmes JEP, Tregastel, France, june 1994, pp. 345-350.

[6] KAPD, King Abdul Aziz Phonatic Database.

[7] – الغامدي، منصور بن محمد (1421هـ) الصوتيات العربية، ص 10-30 مكتبة التوبة، الرياض.

[8] Saud B. Al-Barrak & Mshari A. Al-Jubair, "Arabic Speech Analysis & Recognition Using Neural Network".

[9] L.R. Rabiner and B. Juang, "Fundamentals of Speech Recognition", Prentice Hall, Englewood Cliffs, New Jersey, USA, 1993.

[10] Andrew Kingston, "Learning By Machine", Technical Report, pp 9-13, October 1992.

[11] Mansor Algamdi, "Automatic Segmentation of Speech", Project, pp. 3-64.

[12] H. E. El Khoury, C. E. Jabra Alagha, J. A., M. M. El Choubassi, Skaf and M. A. Al-Alaoui "Arabic Speech Recognition Using Recurrent Neural Networks", Beirut, PP 4-6,200.

[13] Ossama Essa, "Using Prosody in Automatic Segmentation of Speech", Computer Science Department, University of South Carolina, Columbia.

[14] Steve Cassidy, "Speech Recognition", Sydney Australia,2002, pp. 10-35 Department of Computing.

[15] Iman Abuel Maaly Abdel Rahman, "Arabic Speech Recognition using HMM, September,1997, phd thesis.

[16] J. C. Simon, "Spoken Language Generation and Understanding", proceedings of the NATO advanced study institute, D. Reidel publishing company. pp. 103-123. V59. 1980.

[17] *Howard Demuth,* Mark Beale, "Nueral Network Tool Book, for use with matlab", chapter 2-9.

[18] Soong, F. K., "A Training Procedure For A Segment-Based-Network Approach To Isolated Word Recognition", *Proc. Of The IEEE Int. Conf. On Acoustics, Speech and Signal Processing,* ICASSP '87, pp. 693-696, Dallas, 1987.

[19] Dai, J., MacKenzie, I. G., and Tyler, J. E. M., "Stochastic Modelling Of Temporal Information In Speech For Hidden Markov Models", IEEE Trans. on Speech and Audio Processing , vol. 2, No.1, pp. 102-104, January 1994.

[20] Woodland, P.C., "Isolated Word Speech Recogniton Based On Connectionist Techniques", Br. Telecom. Technol. J. , vol. 8, no.2, pp. 61-66, April 1990.

[21] Burr, D. J., "Experiments On Neural Net Recognition On Spoken And Written Text", *IEEE trans. on Acoustics, Speech and Signal Processing* , vol. 36, no. 7, pp. 1162-1168, July 1988.