تمييز بعض الفونيمات العربيه باستخدام الشبكة العصبية المضببة (ممداني)

م . صبا عبد الواحد صدام

جامعة البصره - كلية العلوم – قسم علوم الحاسبات

الخلاصــة

البحث الحالي يستخدم الشبكة العصبية المضببة (ممداني) لتمييز بعض الفونيمات العربيه, حيث يعرض تطوير لمعمارية الشبكة العصبية المضببة من خلال تقليل عدد ارتباطات الشبكة إلى عدد دوال الانتماء المستخدمة زائداً واحد بدلاً من عدد دوال الانتماء مرفوعة لعدد مدخلات الشبكة. إضافة إلى تقليل زمن تدريب الشبكة , استخدمنا بعض المقاطع الصوتيه المتكونه من فونيمين(صحيح+عله) او ثلاث فونيمات(صحيح + عله + صحيح) ومسجله من قبل اكثر من شخص بمختلف الاعمار ومن كلا الجنسين وبنفس البيئه وباستخدام نفس برنامج التسجيل sound forge 5.0 للتدريب. تم اختبار التقنية المقترحة على مجموعة مختلفة من المقاطع الصوتيه لنفس الفونيمات ومسجله بنفس المواصفات والبيئه والبرنامج المستخدم للتسجيل وقد حصلنا على نسبة تمييز 87%.

الكلمات الدالة: تمييز الكلام , التقاطع الصفري , الشبكات العصبية المضببة الممداني، خوارزمية التعليم المتكيفة.

Recognize some of Arabic phonemes by using Mamdani NeuroFuzzy Network

Saba Abdul-Wahed S.
Basrah Univ., Collage Since, Computer Dept.

Abstract

Modified mamdani neurofuzzy network scheme has been proposed and presented in this paper to recognize some of Arabic phonemes. The number of connecting is reduced to be equal to the number of member ship function plus one. to training this network ,we used some of sound segments which contain from two or three phonemes and recorded from more than one person in different age and both sexes in the same environment by used the same recorded programming (sound forge 5.0) . The proposed technique has been testing by using different group from sound segments which recorded in the same environment and recorded program. The percentage of recognition reaches to 87% to all recorded phonemes

Keywords: Mamdani neurofuzzy, Adaptive back-propagation algorithm, zero crossing, speech recognizing .

1.Introduction

Speech recognition also known as automatic speech recognition or computer speech recognition converts spoken words to machine-readable input for example to key presses, using the binary code for a string of character codes. The term "voice recognition" is sometimes used to refer to speech recognition where the recognition system is trained to a particular speaker as is the case for most desktop recognition software; hence there is an aspect of speaker recognition, which attempts to identify the person speaking, to better recognize what is being said. Speech recognition is a broad term which means it can recognize almost anybody's speech - such as a call centre system designed to recognize many voices. Voice recognition is a system trained to a particular user, where it recognizes their speech based on their unique vocal sound. A fuzzy neural network **or** neurofuzzy system is a learning machine that finds the parameters of a fuzzy system (i.e., fuzzy sets, fuzzy rules) by exploiting approximation techniques from *networks.* Both neural networks and fuzzy systems have some things in common. They can be used for solving a problem (e.g. pattern recognition, regression or density estimation) if there does not exist any mathematical model of the given problem[1,2].

The outline of this paper is as follow: the speech recognition will be discussed in section 2, section 3 will discussed the mamdani neuro fuzzy structure and the learning algorithm, section 4 will contine the data set used in training and testing, section 5 will describe the implementation of NF, section 6 will describe the results and section 7 will be the discussion and conclusions.

2. Speech Recognition

The speech signal is a high redundant and non-stationary signal. This attributes causes the speech signal to be very challenging to work . The speech recognition systems fall into two categories according to [2,3]:

1. Speaker dependent systems that are used and often trained by one person.
2. Speaker independent systems that can be used by anyone.

In general, speaker recognition can be subdivided into speaker identification (who is speaking?) and speaker verification (Is the speaker who we think he or she is?). In addition, speaker identification can be closed-set (The speaker is always one of a closed set used for training.) or open-set (speakers from outside the training set may be examined.).

2.1 Recording the Speech Samples

The first step in the research was building a data base of speech samples. we choice two groups of persons from both sex in different age to implement this work through recording some of segments sound. Sound Forge 5.0 Recorder program was used for recording and processing the speech signals, this program allows us to cancel noise in the signal which may have come from environment noise or sensitivity of the microphones. After using this computer program, we obtain a sound signals that is as pure as possible.

2.2 Segmentation Signal

The second step is to identify the precense of a speech signal. This task is easy if the signal is clear, in this step we assume the signal recorded in best environment (the percentage of noise from microphone is smaller and no fan running in the room).

Zero-crossing rate is an important parameter for voiced/unvoiced classification. It is also often used as a part of the front-end processing in automatic speech recognition system. The zero crossing count is an indicator of the frequency at which the energy is concentrated in the signal spectrum. Voiced speech is produced because of excitation of vocal tract by the periodic flow of air at the glottis and usually shows a low zero-crossing count, whereas the unvoiced speech is produced by the constriction of the vocal tract narrow enough to cause turbulent airflow which results in noise and shows high zero-crossing count. In the context of discrete-time signals, a zero crossing is said to occur if successive samples have different algebraic signs. The rate at which zero crossings occur is a simple measure of the frequency content of a signal. Zero-crossing rate is a measure of number of times in a given time interval/frame that the amplitude of the speech signals passes through a value of zero[4], as show in Fig 1.

The zero crossing method was used in this research to determine the begging and the end of each phoneme in each recorded signal speech through[5]:-

1 .***Calculate the number of the change the sign signal from positive to negative or inverse round the zero axis to each frame(frame containe128 points).***
2. ***Calculate the absolute deferential between each two sequential frames.***
3. ***The maximum from them will be the bester frame to segment the signal.***

2.3 Obtaining Signal Speech Samples[2,5,6]

The next important step in the processing of the signal is to obtain a frequency spectrum of each frame. The purpose of the frequency spectrum is to identify the *formants*, which are the peaks in the frequency spectrum.

One method to obtain a frequency spectrum is to apply the Fast Fourier Transform(FFT) to each frame. The FFT with a million points are common in many applications. Modern signal and image processing applications would be impossible without an efficient method for computing the fast Fourier transform which transform time or space – based data into frequency – based data.

The FFT was used as a features extractor because the frequency magnitude does contain information about the pitch and the formants. Beside the spectral magnitude also holds a great deal of other information beside the pitch and the formants magnitude.

The resulting information can be examined manually to find the best features, and we can obtained them from the signal by done this steps :-

1. ***Run the FFT on frame (the frame content 128 point from signal wave)***
2. ***Calculate the increment value in the frequency($Freq._{inc}$) for each samples using Eq.(1):***

$$Freq._{inc} = \text{Sampling Freq.} / \text{FFT points} \quad \ldots\ldots\ldots(1)$$

3. ***Calculate the frequency at each point ($Freq._{P}$) according to Eq(2):***

$$Freq._{P} = Freq._{inc} * \text{Point.seq} \quad \ldots\ldots\ldots.(2)$$

4. ***Sort descending all the frequencies and its locations in the frame .***

5. ***Select first three values from freq and its location only and save there in files .***
6. ***Repate the steps from step 1 to step 5 until reach process to end of file wave.*** Appendix(1) show the output result of the above steps.

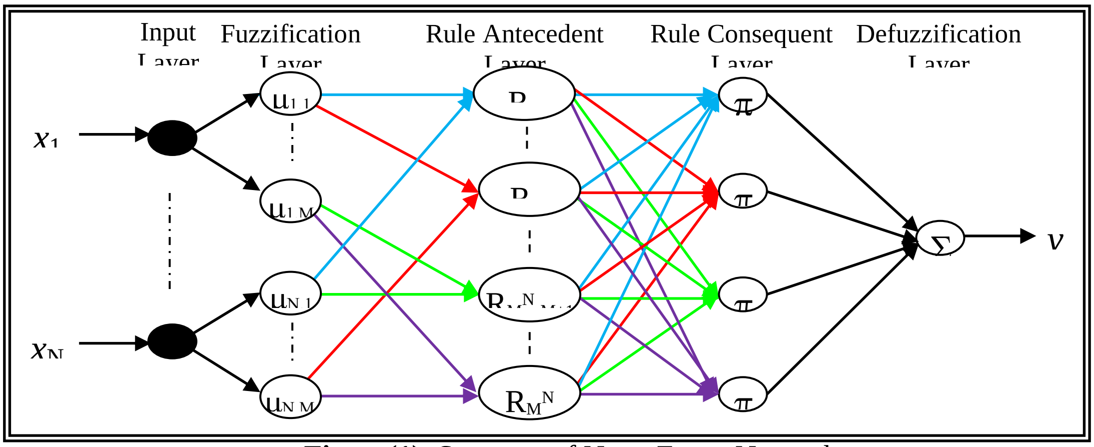
3. NeuroFuzzy Network

The last step in the speech signal processing is to recognize the recorded speech signal. Neural network (NN), Fuzzy network(FN), Neuro fuzzy network(NF), Wavlete, Hidden markove model(HMM) and others methods are used to recognize these signals .

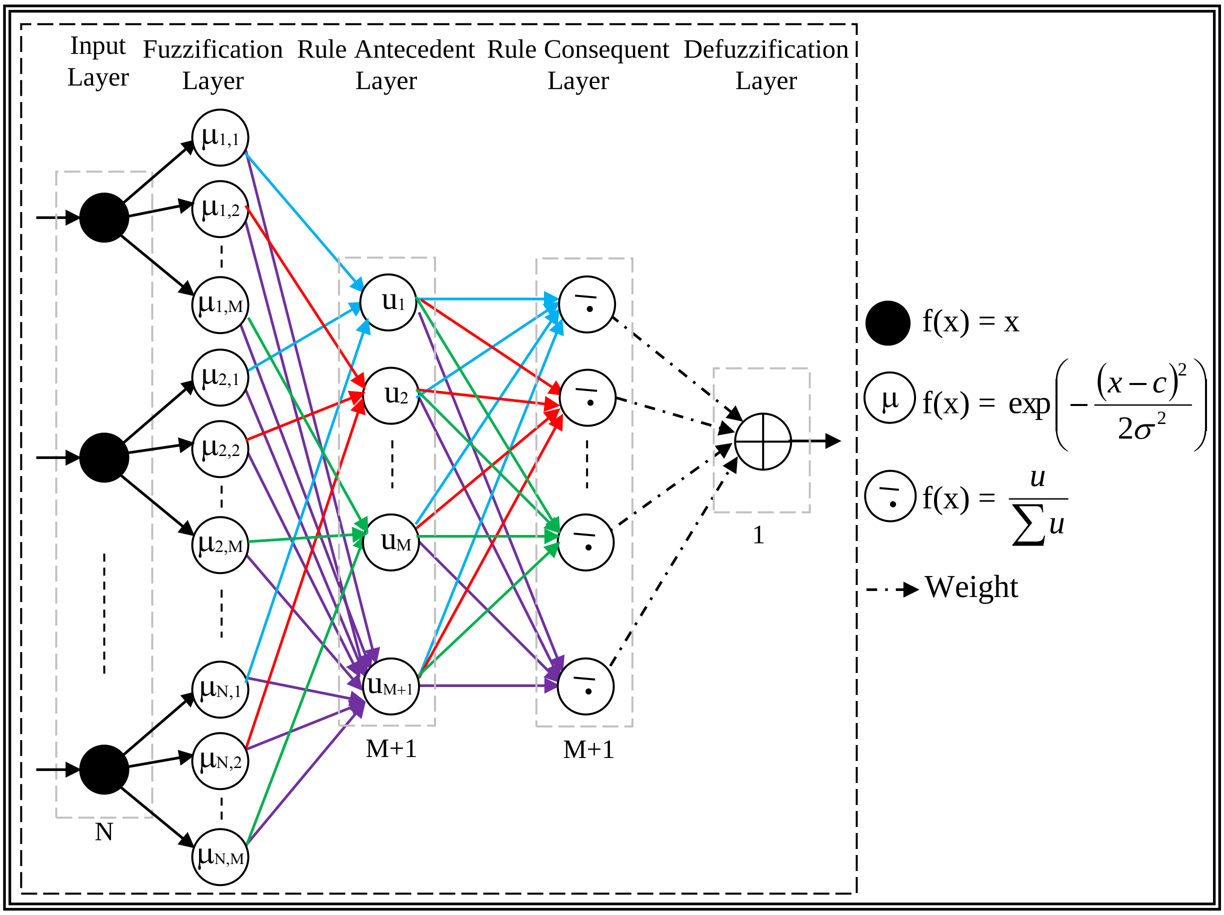
Neuro fuzzy network (NF) was used in this research to recognize some of Arabic phonemes signals. The architecture of the NeuroFuzzy (NF) network based on Mamdani fuzzy inference system consist of five layers; they represent an input layer, fuzzification layer, rule antecedent layer, rule consequent layer, and combination and defuzzification layer respectively. Figure (1) shows the structure of NF network [6-9].

The proposed architecture is based on Mamdani NF which has five layers and three inputs. The scheme has one output and the number of membership functions is generated randomly at each run in the range between 7 and 13.

In common Mamdani NF architecture, full connections are used to connect the neurons of fuzzification layer (membership functions). Thus, the number of connections is equal to number of membership functions powered to the number of inputs. i.e. number of connections is 343 when the number of membership functions is 7 and 2197 when it is 13. As noticed, the number of connections is huge and that will cause impossible learning in training. A suggested modification to the NF architecture can alleviate this drawback, where only the neurons that facing each other were connected and additional one that connects all the neurons of fuzzification layer together. Thus, due to this only 14 connections as maximum are used. The schematic diagram of the NF architecture with our suggested modification is shown in Figure (2).



Figure(1): Structure of NeuroFuzzy Network.



Figure(2): Schematic Diagram of the modified NF Architecture

In order to derive a learning algorithm for a NF network with a gradient descent technique, the inference rule must use differentiable membership function type, for example in this work the Gaussian membership function will be used.

The adjusted parameters in the NF network can be divided into two categories based on ***if*** (antecedent) part and ***then*** (consequent) part of the fuzzy rules. For example in the antecedent part, the mean and variance are fine-tuned, whereas in the consequent part, the adjusted parameters are the consequence weights.

The gradient descent based on Back-propagation (BP) algorithm is employed to adjust the parameters in NF network by using training patterns. Moreover, the algorithm which is used for NF architecture is explained, both feed forward phase and the backward phase of errors.

❖ Forward Phase

This phase computes the activation values of all the nodes in the network from the first to fifth layers.

1. ***Input layer*:** The nodes in this layer only transmit input values (crisp values) to the next layer directly without modification. Thus (Eq.(3)),

$$Net_i = x_i \qquad \forall i = 1..N \qquad \textbf{. . . (3)}$$

Where, N^{1} is a number of neurons in the input layer.

2. ***Fuzzification layer*:** The output function of this node is the degree that the input belongs to the given membership function. Hence, this layer acts as the fuzzifier. Each membership function is Gaussian and an input signal activates only M neighboring membership functions simultaneously. For a Gaussian-shaped membership function, the activation function for each node is as in Eq. (4):

$$\mu_{(i,j)} = \exp\left(-\left(Net_i - C_{(i,j)}\right)^2 \Big/ 2\times\sigma^2_{(i,j)}\right) \qquad \forall i = 1..N \;\; and \;\; j = 1..M \qquad \textbf{. . . (4)}$$

3. ***Rule antecedent layer*:** The implication method is performed by this layer and applied using the products. Where the output of each node is calculated by in Eq.(5):

$$u_{M^{(i-1)}+j} = \prod_{j=1}^{M}\prod_{i=1}^{N}\mu_{(i,j)} \qquad \forall i = 1..N \;\; and \;\; j = 1..M \qquad \textbf{. . . (5)}$$

4. ***Rule consequent layer*:** The normalization process is applied in this layer by implementing Eq. (6).

$$\bar{u}_k = \frac{u_k}{\sum_{k=1}^{M^N} u_k} \qquad \forall k = 1..M^N \qquad \textbf{. . . (6)}$$

5. ***Combination and Defuzzification layer*:** This layer performs defuzzification to produce a crisp output value. Among the commonly used defuzzification strategies, the center of gravity (COG) method yielded the best result. In this layer, linear output activation function is used Eq.(7)

$$y = \sum_{k=1}^{M^N} \bar{u}_k \times w_k \qquad \textbf{. . . (7)}$$

❖ Backward Phase

The goal of this phase is to minimize the error which is calculated as in Eq.(8).

$$E = 0.5 \times (y - d)^2 \qquad \text{... } \mathbf{(8)}$$

Where, *d* is the desired output.

The learning algorithm in NF is realized by adjusting connection weights of the neurons beside the centers and widths of membership functions. Thus, the adaptation of weights of neurons is calculated using Eq. (9-11).

$$w_k^{new} = w_k^{old} + \Delta w_k \quad \forall k = 1..M^N \qquad \text{... } \mathbf{(9)}$$

Where,

$$\Delta w_k = -\eta \times \delta_k^{\,w} \qquad \text{... } \mathbf{(10)}$$

$$\delta_k^{\,w} = \frac{\partial E}{\partial w} = \frac{\partial E}{\partial y} \times \frac{\partial y}{\partial w} = (y - d) \times \frac{u_k}{\sum_{k=1}^{M^N} u_k} \qquad \text{... } \mathbf{(11)}$$

While, adaptation of centers and widths of membership functions is calculated by Eq. (12,14, 15) for centers and by Eq. (13, 16,17) for widths.

$$c_{(i,j)}^{\,new} = c_{(i,j)}^{\,old} + \Delta c_{(i,j)} \quad \forall i = 1..N \;\; and \;\; j = 1..M \qquad \text{... } \mathbf{(12)}$$

And

$$\sigma_{(i,j)}^{\,new} = \sigma_{(i,j)}^{\,old} + \Delta \sigma_{(i,j)} \quad \forall i = 1..\,N \;\; and \;\; j = 1..\,M \qquad \text{... } \mathbf{(13)}$$

Where,

$$\Delta c_{(i,j)} = -\eta \cdot \delta_{(i,j)}^{c} \qquad \text{... } \mathbf{(14)}$$

$$\delta^{c}_{(i,j)} = \frac{\partial E}{\partial y}\times\frac{\partial y}{\partial u}\times\frac{\partial u}{\partial \mu}\times\frac{\partial \mu}{\partial c} = \frac{\left(2\times(y-d)\times(w_k-y)\times u_k\times\left(Net_i - c_{(i,j)}\right)\right)}{\left(\sigma^2_{(i,j)}\times\sum_{k=1}^{M^N}u_k\right)} \quad \textbf{. . . (15)}$$

And

$$\Delta\sigma_{(i,j)} = -\eta\times\delta^{\sigma}_{(i,j)} \quad \textbf{. . . (16)}$$

$$\delta^{\sigma}_{(i,j)} = \frac{\partial E}{\partial y}\times\frac{\partial y}{\partial u}\times\frac{\partial u}{\partial \mu}\times\frac{\partial \mu}{\partial \sigma} = \frac{\left(2\times(y-d)\times(w_k-y)\times u_k\times\left(Net_i - c_{(i,j)}\right)^2\right)}{\left(\sigma^3_{(i,j)}\times\sum_{k=1}^{M^N}u_k\right)} \quad \textbf{. . . (17)}$$

There are several parameters to be adjusted by the gradient descent based on BP algorithms. These parameters are mean and variance of Gaussian memberships and the consequence weights. The BP algorithm that its forward and backward phases illustrated in Section 3 will be used in training the NF network taking into consideration the following points.

① Only the facing neurons in the fuzzification layer will be connected together instead of the full connections. Therefore, the output of the nodes in the rule consequent layer (3rd layer) will be calculated as follows:
The output of the nodes except the last one is computed as in Eq.(18):

$$u_j = \prod_{i=1}^{N}\mu_{(i,j)} \qquad \forall j = 1..M \quad \textbf{. . . (18)}$$

Where, N is number of inputs ($N = 3$) and M is number of membership functions.

While, the output of the last node will be calculated as in Eq(5). Therefore, the carrying out of the last two layers of NF network will be achieved by Eq.(19,20).

$$\bar{u}_j = \frac{u_j}{\sum_{j=1}^{M+1}u_j} \qquad \forall j = 1..M+1 \quad \textbf{. . . (19)}$$

and

$$y = \sum_{j=1}^{M+1}\bar{u}_j\times w_j \quad \textbf{. . . (20)}$$

② BP algorithm will be improved by using adaptive learning rate. Therefore, the parameters are updated under the following conditions which are listed as in Eq. (21):

$$\begin{aligned}
&if\ (new\,error / old\ error) > 1.04\ then \\
&\qquad New\,values\,of\ \ parameters\,are\,discarded \\
&\qquad lr = lr \times lr_dec \\
&else \\
&\qquad Update\,Parameters \\
&\qquad if\ \ new\,error < old\ error \\
&\qquad\qquad lr = lr \times lr_inc
\end{aligned} \qquad \textbf{. . . (21)}$$

③Weights will be updated by using Eq.(9). Where, $\partial E / \partial y$ is calculated by Eq.(22).

$$\frac{\partial E}{\partial y} = (y - d) \qquad \textbf{. . . (22)}$$

④ The adaptation of the other parameters (centers and widths of membership functions) are affected by the type of connections. The actual effect gets from the term of centers and widths errors (δ^C and δ^σ). Thus, centers and widths of membership functions are adapted as in Eq.(13) & Eq.(12) respectively. Where, δ^c and δ^σ are calculated by Eq.(23) & Eq.(24).

$$\begin{aligned}
\delta^c_{(i,j)} &= \frac{\partial E}{\partial y} \times \frac{\partial y}{\partial u} \times \frac{\partial u}{\partial \mu} \times \frac{\partial \mu}{\partial c} \\
&= \left(2 \times (y-d) \times (w_j - y) \times \bar{u}_j \times \left(Net_i - c_{(i,j)} \right) \right) \Big/ \sigma^2_{(i,j)} + \\
&\quad \left(2 \times (y-d) \times (w_{M+1} - y) \times \bar{u}_{M+1} \times \left(Net_i - c_{(i,j)} \right) \right) \Big/ \sigma^2_{(i,j)}
\end{aligned} \qquad \textbf{. . . (23)}$$

$\forall i = 1..N\ \ and\ \ \ j = 1..M$

and

$$\begin{aligned}
\delta^\sigma_{(i,j)} &= \frac{\partial E}{\partial y} \times \frac{\partial y}{\partial u} \times \frac{\partial u}{\partial \mu} \times \frac{\partial \mu}{\partial \sigma} \\
&= \left(2 \times (y-d) \times (w_k - y) \times \bar{u}_j \times \left(Net_i - c_{(i,j)} \right)^2 \right) \Big/ \sigma^3_{(i,j)} \\
&\quad + \left(2 \times (y-d) \times (w_{M+1} - y) \times \bar{u}_{M+1} \times \left(Net_i - c_{(i,j)} \right)^2 \right) \Big/ \sigma^3_{(i,j)}
\end{aligned} \qquad \textbf{. . . (24)}$$

$\forall i = 1..N\ and\ \ j = 1..M$

The initial values of each of the learning rate (***lr***), weights, and widths are generated randomly in range between 0 and 1; where, the width of each membership functions of all the inputs are the same while the values of ***lr_inc*** and ***lr_dec*** were set to 1.3 and 0.1 respectively and the initial values of centers of membership functions of each input are set by using Eq. (25).

$$c_{(i,j)} = (j-1)/(M-1) \quad \forall i = 1..N \text{ and } j = 1..M \qquad \textbf{. . . (25)}$$

The training is stopped after 1500 epochs and it is fail if the error is increased for more than five sequence epochs.

4. Data Sets

In order to implement the mamdani NF network, we choose two groups of person from both sex in different age through recording some of segments sound, these segments are recorded in one environment and in same recorded program (sound forge 5). After segmentation and analysis the signal as in section(2) we get 30 patterns each pattern has 6 inputs and 3 outputs (the outputs represent by 0 and 1) and for each phoneme 6 patterns. Table (1) show the output code to each phonemes which recognized in this work and table (2) show some of the data which the NF network will training on its.

Table (1): the code of output.

Phoneme	code
س	100
ش	010
ا	110
و	001
ي	101

Table (2): part of data training.

10508.2	6373.8	7924.2	61	37	46	0	0	1
5684.8	5857.0	6718.4	33	34	39	0	0	1
5857.0	6029.3	6546.1	34	35	38	0	0	1
5857.0	6029.3	6201.6	34	35	36	0	0	1
6201.6	10508.2	8268.8	36	61	48	0	0	1
1033.6	861.3	1205.9	6	5	7	0	1	1
1033.6	344.5	861.3	6	2	5	0	1	1
1033.6	1205.9	689.1	6	7	4	0	1	1
1033.6	861.3	1205.9	6	5	7	0	1	1
1205.9	1550.4	1033.6	7	9	6	0	1	1
344.5	1378.1	861.3	2	8	5	0	1	1

5. The Implementation of NF

The proposed NF scheme will be trained using the training data set that is constructed from the recorded segments as explained previously. The training will be repeated five times and the learning parameters will be kept the same at each time. For each training run, the performance measure which is represented by the error (Eq.(8)) will be listed in Table (3) beside the number of membership function that is generated for each run.

Table (3): Final Performance Measure of the proposed NF scheme.

Training No.	Number of membership functions	Performance
1	23	0.00998
2	24	0.00998
3	23	0.00999
4	28	0.00997
5	24	0.00999
6	21	0.00999
7	22	0.00999
8	23	0.00998
9	29	0.00998
10	23	0.00998

6. Results

The proposed NF scheme is implemented for recognize above group of segments sound contain two or three phonemes sorts as constant- vowel or constant-vowel-constant, each segment recorded three times from more than one person and from both sexs, and, difficulty known in the possibility of separating the largest number of Arabic characters for the convergence of the existing terms of (parts of speech, classified according to the exits, etc ...) this algorithm was able to recognize the three larger vowels phonemes in addition to some of constant phonemes. Table(4) show the percentage recognize to the recorded segment and table(5) show the recognize percentage to the phonemes.

Table(4): the recognize percentage to the some of recorded segments.

Recorded segment	Recognize percentage
سا	95%
شا	90%
ساس	95%
شاس	85%
سوش	80%
شيس	80%
سي	90%

Table(5): the recognize percentage to some of phonemes.

phoneme	Recognize percentage
السين	95%
الشين	90%
الالف	95%
الواو	80%
الياء	85%

7. Discussion and Conclusions

The main objective of this paper is to recognize some of Arabic phonemes using one of the most common techniques in the soft computing is the NF networks that based on Mamdani fuzzy logic system and neural network algorithms.

A modified Mamdani NF architecture is proposed and presented where only the neurons that facing each other will be connected and additional one that connects all the neurons of fuzzification layer together to cover all the possibilities. This is reduced the number of connections to the number of the number of membership functions plus one and the time of computation is reduced.

References

[1] A.vijay Kumar, M.Vijayapal Reddy, "***A FuzzyNeural Network for Speech Recognition*'**,ARPN journal of system**s** and software Vol.1, No.9, December 2011.

[2] Arjuwan M.Abduljawad Al-jawadi, "***Speech Recognition and Retrieving using Fuzzy Logic System"***, Technical college, Foundation of Technical, Engineering , Mosul, Iraq, 2009.

[3] HO, K. Lai, and, Octavian, C., "***Speech Processing Worshop***", Department of Electrical and Electronic Engineering, part IV project Report 2003.

[4] D.S.Shete, Prof. S.B. Patil ,Prof. S.B. Patil," ***Zero Crossing Rate and Energy of the Speech Signal of Devanagari Script*** ",IOSR Journal of VLSI and Signal Processing (IOSR-JVSP) Volume 4, Issue 1, Ver. I (Jan. 2014), PP 01-05.

[5] صبا عبد الواحد صدام ."استخدام الخوارزميه الجينيه لتحديد معماريه شبكه انسياب الخطأ خلفا المعدله بمعامل تعلم متكيف لتمييز بعض الفونيمات العربيه " , مجله ابحاث البصره (العلميات),العدد الرابع والثلاثون الجزء الثاني, الصفحه (7-14), 2008 .

[6] Raidah Salim, Saba Abdul-Wahed, Bydaa Abdul-Qader**,*"Neuro Fuzzy Network To Recognize Some of Arabic Phonemes"***, Basrah University, Collage of Science, Computer Department, 2014.

[7] Aziz S.B. "***Image Filtering Based on Soft Computing Techniques***", PhD thesis, Basrah University , Collage of Science, Computer Department, 2006.

[8] Qin H." ***Nonlinear Adaptive Noise Cancellation for 2-D Signals with Neuro-Fuzzy Inference System***", M.Sc. thesis, Guelph University, 2005.

[9] Koivo H. "***Soft Computing In Dynamical Systems"***, PhD thesis, Helsinki University of Technology, 2001.

Appendix (1)

Frame No	Frequencies			Locations of frequencies			Absolute deferential
1	10508.2	10680.5	10335.9	61	62	60	42
2	10508.2	10852.7	6029.3	61	63	35	0
3	10508.2	6373.8	7924.2	61	37	46	1
4	5684.8	5512.5	10508.2	33	32	61	7
5	5684.8	5857.0	6718.4	33	34	39	3
6	5684.8	5857.0	10508.2	33	34	61	0
7	5857.0	6029.3	6546.1	34	35	38	1
8	5857.0	10508.2	1894.9	34	61	11	0
9	5857.0	6029.3	6201.6	34	35	36	0
10	6029.3	10508.2	5684.8	35	61	33	2
11	6201.6	10508.2	8268.8	36	61	48	1
12	5684.8	5168.0	5512.5	33	30	32	2
13	516.8	344.5	10508.2	3	2	61	10
14	689.1	344.5	516.8	4	2	3	16
15	689.1	344.5	861.3	4	2	5	1
16	689.1	861.3	344.5	4	5	2	2
17	689.1	861.3	344.5	4	5	2	9
18	861.3	689.1	2067.2	5	4	12	9
19	861.3	689.1	516.8	5	4	3	2
20	861.3	689.1	1033.6	5	4	6	3
21	861.3	689.1	1033.6	5	4	6	2
22	861.3	1033.6	1205.9	5	6	7	2
23	861.3	1033.6	689.1	5	6	4	3
24	1033.6	861.3	689.1	6	5	4	2
25	1033.6	861.3	1205.9	6	5	7	1
26	1033.6	861.3	1205.9	6	5	7	2
27	1033.6	344.5	861.3	6	2	5	4
28	1033.6	1205.9	516.8	6	7	3	3
29	1033.6	1205.9	689.1	6	7	4	1
30	1033.6	1205.9	516.8	6	7	3	2
31	1033.6	861.3	1205.9	6	5	7	2
32	1033.6	1205.9	861.3	6	7	5	2
33	1205.9	1550.4	1033.6	7	9	6	0
34	1205.9	1550.4	344.5	7	9	2	0
35	344.5	1378.1	861.3	2	8	5	4
36	344.5	516.8	1033.6	2	3	6	4
37	344.5	516.8	10508.2	2	3	61	2
38	344.5	10508.2	516.8	2	61	3	9
39	344.5	10508.2	1378.1	2	61	8	1
40	10508.2	344.5	11025.0	61	2	64	17
41	10508.2	10680.5	10852.7	61	62	63	8
42	10508.2	10680.5	10335.9	61	62	60	3