Mansoura Engineering Journal (MEJ), Vol.18, No.4, March 1993

استخدام الشبكات العصبية فى تحويل الكلام المنطوق الى نص مكتوب

A Neural Network Based Arabic Speech Filing System

A.S. Tolba[1] and I. I. Ibrahim[2]

1 Department of Electrical Engineering,
University of Suez-Canal, Port-Said, Egypt
2 Department of Electronics and Communications,
University of Helwan, Cairo, Egypt

ملخص البحث:

يتطلب الاتصال الطبيعى بين الانسان والحاسب عند معالجة النصوص اللجوء الى استخدام الكلام لادخال النصوص . لذلك يجب تزويد الحاسب بامكانية تحويل الكلام المنطوق الى نص مكتوب . وهذه الامكانية تتطلب مواجهة العديد من المشاكل مثل الضوضاء واختلاف المتكلم وتغير طبيعة الكلام من وقت لآخر. ويمكن التغلب على هذه المشاكل باستخدام الشبكات العصبية والتى تتميز بامكانية استخلاص الكلمات التى تتبلمها فى وجود ضوضاء قد تشوبها بالاضافة الى أنها قادرة على استخلاص سمات عامة مميزة للكلملات التى ينطقها أكثر من متكلم . يقدم البحث شبكة عصبية مكونة من ثلاث طبقات للتعرف على الكلمة المنطوقة والتى تناظر حرف من حروف اللغة العربية وبعد ذلك يستخدم قاموس للحصول على الحرف المناظر لخرج المصنف. استخدمت طريقة الانتشار الخلفى لتدريب الشبكة على جميع الأرقام العربية التى ينطقها أكثر من متكلم. بعد تدريب الشبكة لمدة ثلاث أيام متواصلة أمكن للشبكة التعرف على الكلمات التى قدمت اليها. قدم البحث أيضا حلقة الوصل بين كل من معالج نصوص عربى - انجليزى والشبكة العصبية الخاصة بتصنيف الكلمات. ويتوقع أن تلعب عملية تحويل الكلام المنطوق الى نص مكتوب دورا هاما فى أتمتة الأعمال المكتبية .

Abstract:

A natural man - machine communication with word processors requires the integration of speech recognition techniques. Speech recognition task is a difficult problem because of its temporal nature and the possibility of the presence of noise. Therefore, it is difficult to extract significant features by using the traditional techniques. Artificial neural networks introduce simple and fast learning techniques for this problem. This paper introduces the application of a three layer neural network for converting spoken Arabic words into Arabic text. We describe a supervised learning method which is based on the well known back propagation technique. The designed network proves itself to be

speaker independent and noise immune. Word spotting guarantees invariance under translation in time. A user friendly word processor should have intelligent interfaces such as the speaker interface and the hand writing interface. In this work we introduce the first interface. It is expected that neural network based speech-to-text transcription systems should have a significant effect on office automation.

1. Introduction

Integration of speech input in a word processing environment would relief the human from tidy typing of letters and introduces a user friendly interface. Achieving this goal is not a simple task due to the variety of technical problems which should have to be solved. The first major problem is that of speaker independent and time invariant speech recognition. The second problem is the segmentation of a stream of spoken words. A third problem is the interfacing of the neural speech classifier with a bilingual word processor.

The application of neural nets for speech classification is still in its infancy phase. New learning algorithms for neural networks have recently appeared [1-6]. Algorithms based on the use of Artificial Neural Networks (ANNs) are applied for solving speech recognition problems such as consonant recognition, vowel recognition, syllable recognition, as well as single word recognition [1-6]. Some of these investigations have proven the superiority of neural nets, compared to the use of other techniques. Burr [1] has performed a number of experiments for assessing the performance of neural networks on recognition of spoken words. Results indicated that neural networks and nearest neighbor classifiers perform at near the same level of accuracy. Bauer and Geisel [2] have showed that multi-layer perceptrons with feedback could be used to alleviate the problems of traditional word recognition techniques. Kowalesky and Strobe [3] have presented a single hidden layer neural network for word recognition. They implemented an error back propagation network for perfect learning of a one speaker vocabulary of 13 words. Their network is robust against velocity variations in the speech data. Plaut and Hinton [4] have described the application of neural networks for learning a set of filters that enable to recognize real speech. The potential of self organizing maps for solving transcription problems was demonstrated on the hand of a neural phonetic typewriter [11].

In this investigation we introduce the first steps towards an Arabic speech filing system. This system includes three major fields:

1. Neural speech classifier
2. Interfacing classified spoken words with an Arabic/Latin word processor
3. Bilingual word processing

In section 2, we begin by introducing the system architecture and show the ability of ANNs for speech pattern recognition.

In section 3, we start by describing the preprocessing phase and then show the ability of ANNs for feature extraction. Next, we describe the back propagation learning algorithm. The neural network design is then described and its capability for pattern classification is shown. In section 4, we then describe the interfacing of the neural speech classifier with a A/L word processor. The results are then described and summarized in the final section of this paper.

2. System Architecture:

Figure 1. shows the block diagram of speech-to-text transcription system. The speech signal is Low Pass Filtered (LPF) with a cut-off frequency of about 5 kHz and then captured by a 8-Bit PCM speech digitizer (ADC) with 15 Kbytes/sec sample rate. The spoken word is then passed to the word spotting procedure to localize the word to be fed into the input layer of the neural network. After training the neural network on an exhaustive training set which represents different varieties of the ten Arabic digits, the weighting coefficients of its synaptic connections are stored in the knowledge base for further classification of an incoming word. The classified digit is then interfaced with the bilingual (Arabic/Latin) word processor [13] which transcribes the word into its corresponding Arabic text (digit or word), which is then stored on a Hard Disk (HD).

3. Speech Classification With An Artificial Neural Network:

3.1 Preprocessing:

The speech signal is first low pass filtered and then spotted through a sliding window to determine both the starting and ending points of a word. Preprocessing retains all significant features of the spoken word. Experiments were conducted to hear the word after its spotting by a sliding window. A set of 60 overlapping sliding windows with overlap segments of 40 samples are used. Each segment includes 2048 samples. The sum of absolute values of the samples in a given window is then computed. Next the window of interest is selected according to the criterion of maximum sum of absolute sample values. Prewhitening is then used for elimination of trend in the spotted word.

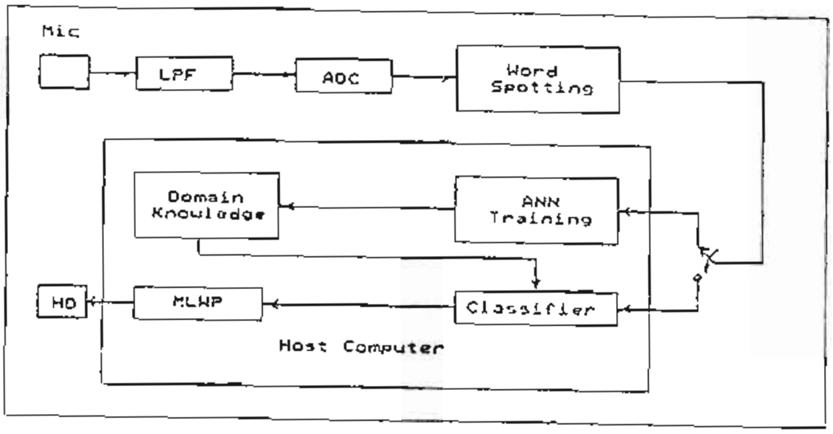


Figure 1. Block Diagram Of Speech-to-Text Transcription System

3.2 Speech Feature Extraction Using Neural Networks:

Training of a multi layer neural network on the hand of an exhaustive training set maps all the features of the given speech signals into the weighting space. The weighting coefficients of the synaptic connections represent the knowledge in the whole training set and we think that these coefficients represent the principal components of the trained signals.

3.3 Neural Network Architecture:

A three layer neural network (Fig. 2) is designed and trained for word classification. The input layer consists of 2048 input neurons. A single hidden layer is proved sufficient for speech learning [1-6]. Two hidden neurons are proved to be sufficient for encoding of speech domain knowledge. The output layer has eleven output units for discrimination of the ten spoken Arabic digits and indication of the nature of word representation in text (1 or one). The neural network perform nonlinear transformation on its summed inputs and produce outputs between 0.142 and 0.99 . The output of the ith unit is computed by summing all of its weighted inputs y_j as follows [7-11]:

$$x_i = \Sigma_j y_j w_{ji} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(1)$$

where w_{ji} is the weight from the jth to the ith unit. The sigmoid function is then applied to the result of this summation:

$$y_i = 1 / (1+e^{(-x_i)}) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(2)$$

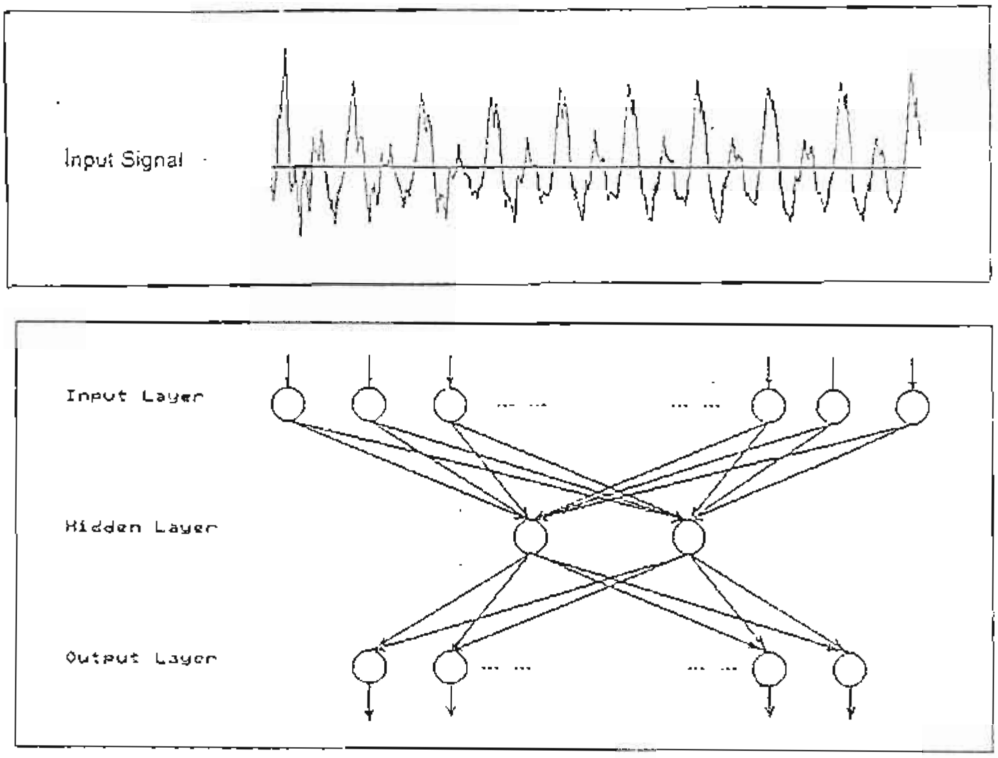


Figure 2. Architecture of the Speech Classifier

The input layer (sensor neurons for data acquisition) of the network is made up of 2048 units, each clamped to a single sample of the spoken word to be learned. The number of output units is set as eleven. The states of the output units represent the class of the word presented to the input layer. An intermediate hidden layer of two neurons allows the network to assign the input pattern to the appropriate output class. The connections between the different neurons measure the degree of correlation between activity levels of neurons they connect.

3.4 Back Propagation Learning Algorithm:

Back propagation is an iterative learning technique whose convergence is highly problem dependent [7,10,14,15]. For each learning cycle, the input layer was clamped to the samples of a single word from the training set. The activity of each unit was propagated forward through each layer of the network using equations (1) and (2). The activity at the output layer was compared to the desired output, and an error for each output unit was calculated as follows:

$$E = 0.5 \, \Sigma_c \, \Sigma_i \, (y_{i,c} - d_{i,c})^2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3)$$

where c is an index over cases (input-output pairs), i is an index over output units, y is the actual state of an output unit, and d is the desired state. The learning procedure minimizes E by performing gradient descent in the weight space [1].

The error term of equation 3 is used for updating weights according to the layer dealt with. The weight update equation for the output layer is defined by [4]

$$w^{o}_{kj}(t+1)=w^{o}_{kj}(t)+\alpha\delta^{o}_{pk}i_{pj} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4)$$

The hidden layer weights are updated according to the following equation [4]

$$w^{h}_{ji}(t+1)=w^{h}_{ji}(t)+\alpha\delta^{h}_{pj}\, x_{i} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(5)$$

The effect of different variables on the learning time is described in detail in [12].

4. Arabic Word Processing:

Arabization of word processors represents one of the basic milestones towards reducing the fears of computers as a result of language barriers. Any computer application that handles Arabic text must confront two main classes of problems [1-2]. The first class is caused by the cursive interacting nature of the Arabic letters. The computer must take a sequence of isolated Arabic characters and then connect them according to the context in which they are written. This process is called contextual formatting. Another class of problems is the directional layout which encounters the right positioning of Arabic and European languages on the same line.

A bilingual word processor (BLWP) that can intermix right-to-left Arabic-text with left-to-right text in European languages is developed. Very complex algorithms are developed for both contextual formatting and directional layout. BLWP is a self contained program that is independent of any other display fonts and generates its own printer fonts which are down loadable to the buffer in the attached printer. The most important feature of this word processor is its independence of the hardware.

4.1 Toward a More Friendly User-Interface:

Conventional word processors usually have two user interfaces at least: the keyboard and the screen. In this paper, interfacing is extended to include the speaker, the keyboard and the screen. The interfacing of the speech classifier with the bilingual word processor is implemented with a dictionary. This dictionary has the following three entries:

1. Classifier output,
2. Arabic/Latin word,
3. Arabic/Latin digit.

The performance of the speech-to-text transcriptor is important and must be fast enough to avoid frustration. The structure of the dictionary is thus of great importance. It must allow very fast searches. Algorithms and data structures for speech-to-text transcription must be tailored to the properties of the language to be processed. Investigation of these areas is going on.

4.2 Principal of Work of Speech-to-Text Transcriptor :

The working principal of a speech-to-text transcriptor could be summarized in the following sequence of pseudo code:

```
Loop:
 Interpretation of either a keyboard character  or a speaker word
 Processing  a character,  or transcription  of a word to  Arabic
 text
 Show modification
End of loop
Save text
```

5. Experiments and Results:

After preprocessing the signals of the words of the training set, word spotting is then performed. The criterion for word spotting is based on the maximum sum of the absolute sample values in 60 overlapping windows. The resulting words are then fed randomly into the input layer of the neural network. The network is trained after 330,000 cycles using the back propagation technique. All digits are correctly classified after two days of continuous training on a 50 MHz, 80486DX2 machine. Training the network on both the basic Arabic alphabet (28 words) and the 26 words of the Latin alphabet is going on. The generalization of the network shows its capability for speaker independent word recognition. Figure 3 shows the signals of ten spoken Arabic digits used in the study, and the corresponding dictionary entries. A keyword (spoken SHIFT) is used to determine if the dictated word is to be translated as a digit or a text word. This means that a spoken word is classified as a digit according to the context of its previous word.

6. Conclusions:

This paper presents research work towards the development of an intelligent Arabic/Latin word processor. An interfacing with a bilingual word processor is implemented. The applicability of a multi layer neural network in the design of an Arabic Speech-to-Text filling system is proved to be successful. Preliminary research showed encouraging results. We have used a two layer network for training and speaker independent classification of ten spoken digits. We intend to complete the training of the neural network to classify all the necessary spoken Arabic and English alphabet characters.

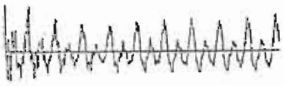
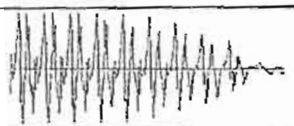
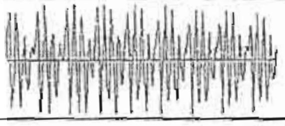
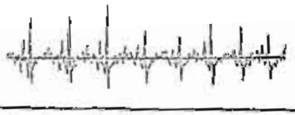
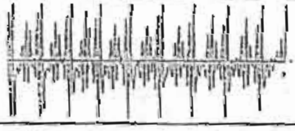
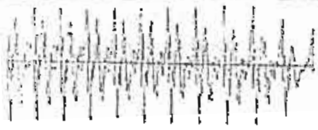
Classifier output	Speech Signal	Digit	Word
0.14		1	واحد
0.23		2	اثنين
0.31		3	ثلاثة
0.40		4	اربعة
0.48		5	خمسة
0.57		6	ستة
0.65		7	سبعة
0.74		8	ثمانية
0.82		9	تسعة
0.91		10	عشرة
0.99			shift

Figure 3. Speech Signals for ten spoken Arabic digits and SHIFT

7. References

[1] D. J. Burr, " Experiments On Neural Net Recognition of Spoken and Written Text ", in IEEE Trans. On. ASSP, VOL. 36, NO. 7, pp. 1162-1168, JULY 1988.

[2] H. U. Bauer and T. Geisel, " Sequence Analysis and Feedback Multi-layer Perceptrons ", in Parallel Processing in Neural Systems and Computers, R. Eckmuiller, G. Hartmann and G. Huaske (Editors), Elsievier Science Publishers B.V. (North Holland), pp. 375-382, 1990.

[3] F. Kowalewsky and H. W. Strobe, " Word Recognition with a recurrent Neural Network", in Parallel Processing in Neural Systems and Computers, R. Eckmuiller, G. Hartmann and G. Huaske (Editors), Elsievier Science Publishers B.V. (North Holland), pp. 391-394, 1990.

[4] D. C. Plaut and G.E. Hinton, "Learning Sets of filters using Back Propagation", in Computer-Speech and Language VOL. 2, pp. 35-61, 1987.

[5] R. B. Melton and R. S. Barga, " Neural Networks and data pipelines", through personal contacts.

[6] R. S. Barga, M. A. Friesel and R.B. Melton,"Classification of acoustic emission waveforms for nondestructive evaluation using neural networks", through personal contacts.

[7] B. Mueller and J. Renhardt, "Neural Networks: An Introducetion",Springer-Verlag, Berlin,Prentice Hall Inc., 1992.

[8] B. Kosoko,"Neural Networks and Fuzzy Systems: A dynamical Systems approach to machine intelligence", Prentice Hall International Inc., 1992.

[9] J. Herz, A. Krogh, and R. G. Palmer, "Introduction To The Theory Of the Neural Computation" Addison-Wesley Publishing Company, New York, 1991.

[10] R. P. Lipmann, "An Introduction to computing with Neural Nets" , IEEE ASSP Magazine, pp. 4-22, April 1987.

[11] A. J. Freeman and D. M. Skapura "Neural Networks: Algorithms, Applications, and Programming Techniques", Addison-Wesley Publishing Company, 1991.

[12] A. S. Tolba, "A Neural Network For Arabic Character Recognition", to be published.

[13] A. S. Tolba, "A multilingual Word Processor", to be published.

[14] F. F. Solie, P.G. Gallinari and S. Thiria, " Learning and Associative Memory", Pattern Recognition Theory and Applications , Edited by Pierre A. Devijier Kittler, NATO ASI series, Springer Verlag, pp. 249-267, 1986.

[15] R. Braham and J.O. Hamblen, "The Design Of A Neural Network With A Biologically Motivated architecture", IEEE Trans. on Neural Networks , Vol 1. No.3 ,pp. 251-261, September 1990.