# Speech Analysis Tools As Back-Ends for Bangla Phoneme Recognition Using MFCC, Neural Network, Hamming and Euclidean Distance

Md. Atiqul Islam [a], Nur Hossain Khan [b *], Md. Hasinur Rahman [c], Md. Abdus Satter [d]

[a] Programmer, Sonali Bank Limited, Bangladesh
[b] Assistant Maintenance Engineer, Bangladesh Bank, Bangladesh
[c] Assistant Manager, Meghna Petroleum Limited, Bangladesh
[d] Senior Officer, Agrani Bank Limited, Bangladesh

**Article Info**

Article history:
Received 5 February 2015
Received in revised form
10 February 2015
Accepted 28 February 2015
Available online 15 March 2015

**Keywords**

Speech recognition,
Bangla phonemes,
MFCC,
Neural network,
Hamming distance,
Euclidean distance

**Abstract**

This paper deals with the development of a speech recognition system and comparative study of recognition results for Bangla phonemes. At first, Phonemes were recorded and converted into digital form. Then MFCC features from phonemes were extracted by Mel scale cepstral analysis. The recognition tools include Hamming and Euclidean distance measurement and learning through a neural network. Ten Bangla phonemes were used to test the system. The performance of the system shows that Euclidean distance measurement is the simplest and better method in recognizing Bangla phonemes.

## 1. Introduction

Speech recognition is a formidable problem; many approaches have been tried, with only mild success. This is an active area of DSP research, and will undoubtedly remain so far many years to come. The human vocal cord vibrates with a frequency ranging from 50Hz to 1000Hz during speech production. This is the fundamental frequency and speaker's sex, age, stress, emotion etc. cause the variation. Furthermore, the vocal tract which functions as resonator during speech production is about 17 cm for adult and this length is also variable depending on the speaker [1]. This variation in vocal tract causes variation in resonance frequencies. These problems contribute a major part in complexities in speech recognition. So the first target is assigned for a speech researcher is to find features within a phoneme, which will be almost stable against all the variability. And off course, final requirement is to select a better method, which can recognize phonemes correctly using these features. During a few decades researchers are continuing search for better methods and techniques that might bring us closer to accurate recognition. In this search for better method, two distance measurement techniques (Hamming and Euclidean) and neural network are used in our study.

## 2. Distance Measurement Techniques

Several distance measurement techniques are used in pattern recognition. Two of them are discussed below. The most basic measure and one that is widely used because of its simplicity is the *Hamming Distance* measure. For the two vectors

$$X = (x_1, x_2, ...............x_N)$$

$$Y = (y_1, y_2, ..............x_N)$$

The Hamming distance is found by summing the absolute difference between the corresponding components as

$$H = \sum_{i=0}^{N} \left( |x_i - y_i| \right)$$

Another most common methods used is the *Euclidean Distance* measure. For the two vectors X and Y as defined above, the shortest distance is the Euclidean Distance that is defined by:

$$d(X,Y)_{euc} = \sqrt{\left( \sum_{i=1}^{N} \left( X_i - Y_i \right)^2 \right)}$$

Where N is the dimensionality of the vectors

## 3. Artificial Neural Network

Artificial neural networks, specially the multi-layer feed forward neural networks have long been used in the field of pattern learning for their competence in learning. The structure of such network is shown in figure-1. It consists of one input layer, one output layer and one hidden layer between them. In our study, the input layer contents the source nodes that provide physical access points to the input speech. The hidden and output layers consist of computation nodes. The input neurons simply distribute the speech data along multiple paths to the hidden layer neurons. A weight is associated with each connection to hidden neurons. The part of speech presented to the hidden layer neuron due to one single connection is the product of output value of input node and the connected weight. Among the learning algorithms for this feed forward network, the best known is the Error Back propagation (BP) algorithm [2]. The BP algorithm is an iterative gradient algorithm to minimize the Mean Squire Error (MSE) between the actual output and target output.

**Corresponding Author,**
**E-mail address:** nur_cse_iu@yahoo.com

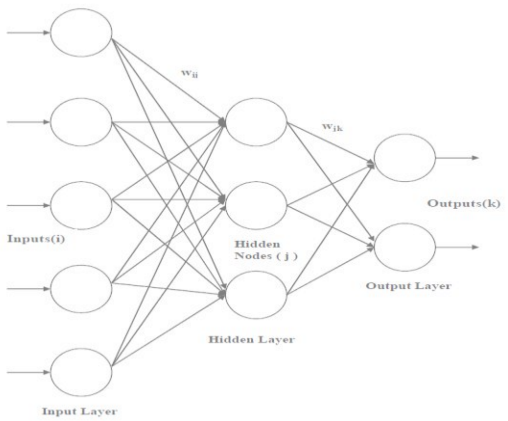

**Fig: 1.** The Basic Model of Multilayer Feed forward

The Back propagation learning algorithm is given below:

**a. Initialize weights and thresholds**

Set all weights and thresholds to small random values.

**b. Present input and desired output**

Present input as $x_i$ and target output as $t_i$ .

**c. Calculate actual output**

Using sigmoid function of the from $f(x) = 1/1+\exp(-\lambda x_i)$ , each node calculates

$$y_j = \frac{1}{1+\exp\left[-\lambda\left(\sum_{i=1}^{n} w_{ij} y_i^1 - \theta\right)\right]}$$

Where $y_i$ is the output of the node in consideration [3, 4],

$y_i^1$ is the output of the nodes of the previous layer - $x_i$ in case of first layer,

$w_{ij}$ is the weight for i-j connection

$\theta_j$ is the threshold for the node in consideration

$\lambda$ Determines the slop of the function at point and is called the activation gain. In our program, two values are used for it, one is *spr*. and other is *spread*.

**d. Adapt weights and thresholds**

Start from the output layer and work backwards. The weight correction for the hidden-to-output weight matrix elements [5, 6]

$$\Delta w_{jk} = \eta_2 (t_k - y_k)(1 - y_k) y_k$$

$$w_{jk}(t+1) = w_{jk}(t) + \Delta w_{jk}$$

And the weight correction for the input-to-hidden weight matrix elements [5, 6]

$$\Delta w_{ij} = \lambda \eta_1 (1 - y_k) y_k \sum_k (t_k - y_k) w_{jk}$$

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}$$

Similarly, correction for thresholds for the hidden-to-output

$$\Delta th_j = \eta_1 (t_k - y_k)(1 - y_k) y_k$$

$$th_k(t+1) = th_k(t) + \Delta th_j$$

And correction for thresholds for the input-to-hidden

$$\Delta th_j = \lambda \eta_1 (1 - y_j) y_j \sum_k (t_k - y_k) w_{jk}$$

$$th_j(t+1) = th_j(t) + \Delta th_j$$

Where $\eta_1$ and $\eta_2$ are small proportionality constant known as learning rate for input to hidden layer and hidden to output layer respectively. The value of $\lambda$ is *spread* as before. $t_k$ denotes target output.

## 4. Computation of Mel Frequency Cepstrum Coefficients (MFCC)

The sequential steps to compute MFCC features from phonemes are shown in figure-2. After preprocessing (Windowing and Preemphasis), the system calculates the DFT using most efficient FFT algorithm. A set of critical band filters evenly spaced along the mel-scale smoothes and averages the FFTed signal into a smaller number of coefficients. Taking the log of each coefficient will force the signal to be minimum phase. The discrete cosine transform can then be used to derive the mel frequency cepstral coefficients (MFCC).

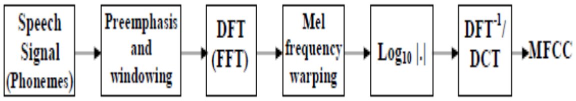

**Fig: 2:** The Sequence of Operations to Convert a Phoneme into a set of MFCC Features

## 5. The Developed Software System

A software system was developed to implement the algorithms discussed above. The programs are written in C language and compiled with Turbo C++ compiler. The block diagram approach of the program is shown in figure-3. Some of the modules of the software were prepared as a moderated version of that in [5] and [7].

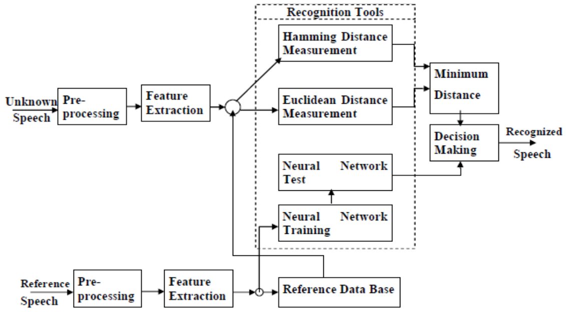

**Fig: 3:** The Main Blocks of the Developed Software

**Table: 1:** The Target Output Pattern

| Phonemes | Node#0 | Node#1 | Node#2 | Node#3 | Node#4 | Node#5 | Node#6 | Node#7 | Node#8 | Node#9 |
|---|---|---|---|---|---|---|---|---|---|---|
| **A** | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Av** | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **B** | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **D** | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| **G** | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| **I** | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| **K** | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| **U** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| **g** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| **k** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Table: 2:** Results of Experiment with Recognition Methods

| Phonemes | Total No. of Utterances | Recognized using Hamming Distance | Recognized using Euclidean Distance | Recognized using Neural Network |
|---|---|---|---|---|
| **A** | 30 | 22 | 22 | 23 |
| **Av** | 30 | 30 | 30 | 30 |
| **B** | 30 | 30 | 30 | 30 |
| **D** | 30 | 30 | 30 | 30 |
| **G** | 30 | 30 | 30 | 30 |
| **I** | 30 | 30 | 30 | 30 |
| **K** | 30 | 23 | 24 | 14 |
| **U** | 30 | 27 | 27 | 29 |
| **g** | 30 | 28 | 28 | 30 |
| **k** | 30 | 30 | 30 | 30 |
| Total | 300 | 280 | 281 | 276 |
| percent | - | 93.33% | 93.66% | 92% |

## 6. Experiment with Recognition Methods

In our experiment with Bangla phonemes, six vowels (/A/, /Av/, /B/, /D/, /G/ and /I/) and four consonants (/K/, /U/, /g/ and /k/) were taken as input to the system. Three hundred utterances of ten phonemes were included and the utterances were recorded in three different time, one week later from the previous one. The duration of each phoneme was 0.5 sec. During computation of MFCC, phoneme data was grouped in a set of samples, called a frame representing 16 msec of speech. A total of 248 feature data are collected from the computation of 8-MFCC per frame. Thus the number of input nodes in the neural network was 248 and 10 output nodes were taken to represent 10 phonemes. The target output pattern is given in table-I. Net parameters used in this experiment were as below:

Hidden Unit=20, spr. =0.25, spread=0.25, eta1=0.7, eta2=0.1

The results of this experiment are given in table-II. As seen, all recognition methods perform nearly the same, Neural Network = 92%, Hamming = 93.33% and Euclidean=93.66%.

## 7. Discussion And Conclusion

Three methods were used in this recognition scheme. They are Neural Network, Hamming and Euclidean distance measurement. As discussed in section-2 and 3, the distance measurements were very simple in computation but neural networks are somewhat more complex. Comparative recognition results were shown in table-2. As seen in table-II, the all methods produce almost the same result, but Euclidean distance produces slightly better result. Because of these and for the simplicity of calculation, Euclidean distance measurement may be concluded as the better method for Bangla phoneme recognition.

## References

[1] Jean-Claude Junqua, Jean-Paul Haton, Robustness in Automatic Speech Recognition: Fundamentals and Applications, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1997

[2] D. E. Rumelhart, J. L. McClelland, The PDP Research Group, Parallel Distributed Processing: Explorations in the Microstructure of Cognition, MIT press, Cambridge, MA, 1, 1986

[3] N. K. Bose, P. Liang, Neural Network Fundamentals: Graphs, Algorithms and applications, Tata McGraw Hill, New Delhi, 1996

[4] Stamatios V. Kartalopoulos, Understanding Neural Networks and Fuzzy Logic: Basic concepts and Application, IEEE, New Delhi, 1996

[5] E. Rietman, Exploring Parellel Processing, Windcrest Books, USA, 1990

[6] R. Ahmed, Pattern Recognition by Neural Network, Department of Applied Physics & Electronics, Rajshahi University, Bangladesh, 1994

[7] Aldebaro Barreto da Rocha Klautau Jr, Old Aldebaro's Home Page, http://speech.ucsd.edu/aldebaro/software.htm, UC San Diego La Jolla, CA 2002

[8] A. Hai, Dhvani-vignan o bangla dhvani tatta. Mullick Brothers, Dhaka, Bangladesh, 1985

[9] M. A. Hasnat, J. Mowla, M. Khan, Isolated and continuous bangla speech recognition: Implementation performance and application perspective, In proceedings of the International Symposium on Natural Language Processing (SNLP), Hanoi, Vietnam, 2007