Bangla User Adaptive Word Speech Recognition: Approaches and Comparisons

Adnan Firoze, Department of Electrical Engineering and Computer Science, North South University, Dhaka, Bangladesh

Md Shamsul Arifin, Department of Electrical Engineering and Computer Science, North South University, Dhaka, Bangladesh

Rashedur M. Rahman, Department of Electrical Engineering and Computer Science, North South University, Dhaka, Bangladesh

ABSTRACT

The paper presents Bangla word speech recognition using two novel approaches with a comprehensive analysis. The first approach is based on spectral analysis and fuzzy logic and the second one uses Mel-Frequency Cepstral Coefficients (MFCC) analysis and feed-forward back-propagation neural networks. As human speech is imprecise and ambiguous, fuzzy logic – the base of which is indeed linguistic ambiguity, could serve as a precise tool for analyzing and recognizing human speech. The authors' systems revolve around the visual representations of voiced signals – the Fourier energy spectrum and the MFCC. The essences of a Fourier energy spectrum and the MFCC are matrices that include information about properties of a sound by storing energy and frequency in discrete time. The decision making process of their systems is based on fuzzy logic and neural networks. Experimental results demonstrate that their fuzzy logic based system is 86% accurate whereas the Artificial Neural Networks (ANN) based system is 90% accurate compared to a commercial Hidden Markov Model (HMM) based speech recognizer that shows 73% accuracy on an average. Moreover, the authors' research derives that, even though ANN gives a better recognition accuracy than the fuzzy logic based system, the fuzzy logic based system is more accurate when it comes to "more difficult" or "polysyllabic" words. In terms of runtime performance, the fuzzy logic based system outperforms the ANN based Bangla speech recognition system.

Keywords: *Artificial Neural Networks (ANN), Backpropagation, Cepstrum, Fuzzy Logic, Melody (MEL) Scale, Mel-Frequency Cepstral Coefficients (MFCC), Segmentation, Spectogram, Speech Recognition, Short-Time Fourier Transform (STFT)*

1. INTRODUCTION

Human speech recognition has a broad elucidation that refers to the technology that can recognize speech. The recognition process is still open because none of the current methods is fast and precise enough compared to human recognition abilities. Research in this area has attracted a great deal of attention over the past five decades. Several technologies have been applied and efforts were made to increase the performance up to marketplace standard so that the users would have the benefit in a number of ways.

During this long research period, several key technologies were applied to recognize isolated words such as Hidden Markov Models

DOI: 10.4018/ijfsa.2013070101

(HMM) (Hasnat, Jabir, & Mumit, 2007), Artificial Neural Networks (ANN), Support Vector Classifiers with HMM, Independent Component Analysis, HMM and Neural-Network Hybrid, the stochastic language model and more (Juang & Rabiner, 2005).

In recognizing Bangla speech, most of the research efforts were developed by HMM and ANN techniques but no research work has been reported till the date that uses Fuzzy logic and develops a Fuzzy Inference System (FIS), MEL (a shortened form of the word 'melody' named by mathematicians – Stevens, Volkman and Newman) filtering and Short-time Fourier transform (STFT) methods. Therefore, in this research we investigate, propose and implement two distinct models that can recognize Bangla isolated words. The first approach is by using spectral analysis and fuzzy logic and the second one is by using cepstral analysis and feed-forward back-propagation artificial neural networks.

The ambiguity in phonemes in Bangla/Bengali speech is more intense and varied than that of English speech since Bangla stems from the 'Indo-European language family' just as Hindi, Urdu, Persian and numerous languages from South Asia having native speakers of over 3 billion (Weiss, 2006). Therefore, the approach for speech recognition considers the 'word level' rather than the 'phonetic level'. In other words, the base or smallest entity of this system is a 'word' (in Bangla) rather than sounds (phonemes) that construct the words. The authors also want to mention that HMM based speech recognizers work from the phonetic level as opposed to a 'word level' since most of the HMM based systems are optimized for English speech.

In the authors' FIS, three inputs have been used, e.g., frequency, energy level of the sample, and the energy level of the target or description. Since human ear is more susceptible to lower frequencies of sounds, FIS rules are made accordingly to put emphasize on the lower frequencies. The output of FIS is the similarity between two 'segments' of a word and the overall evaluation of the FIS has been cumulated to reach the verdict of a word recognition.

However, the authors rely on cepstral analysis for our inputs of the ANN based system. We designed our network using the feed-forward back-propagation architecture with one hidden layer (between the input and output layers) consisting of 60 neurons (the justification is elaborated in subsection 3.10.2). The inputs for our ANN based system are somewhat similar to the inputs of our fuzzy logic based system; however, instead of using spectral analysis directly, we take the energy levels and frequency of audio signals to the paradigm of cepstral analysis. Using Mel-frequency cepstral coefficients (MFCC) as features in the input, we recognize the Bangla words from the outputs of the ANN.

The organization of the paper is as follows: Section 2 discusses the related works done till date in relevance to speech recognition emphasizing on Bangla speech (phoneme descriptions, vowels and recognition systems) in particular. Section 3 presents the detailed descriptions of the strategies that we implement to build our systems. In Section 4 we report and analyze the experimental results. Finally, Section 5 concludes and gives directions of our future research.

2. RELATED WORKS

Even though 'speech recognition' is still an open problem with quite low accuracy, the attempt to recognize speech dates back to the 1950s. The very first speech recognizer only recognized digits that were spoken (Davies, Biddulph, & Balashek, 1952). After the first attempt, the speech recognition researches were centered on voice commands in devices and utility services. In 1990 AT&T call center service devised the first command recognition. Their help-lines facilitated voice commands (Juang & Rabiner, 2005). However, this attempt was not successful since most dialects could not be recognized.

Since then, approaches were revolved around the visual representation of speech.

Documentation of the relationship between a given speech spectrum and its acoustic properties were done in 1922 by Fletcher and others at the Bell Laboratories (Fletcher, 1922). Thus, Harvey Fletcher became one of the pioneers in recognizing the importance of the spectral analysis in detecting phonetic attributes of sound.

Even though the attempts to recognize human speech properly goes a long time back in time, the speech recognition approaches in Bangla language started only in the 21st century. In a research work, Roy, Das, and Ali (2002) performed the recognition by using Back-propagation ANN. A phoneme recognition approach using ANN as a classifier was devised (Hasan, Nath, & Alauddin, 2003). Root Mean Squared (RMS) energy level was calculated by them as features from the filtered digitized signal by them.

Karim, Rahman, Iqbal, and Zafar (2002) presented a technique to recognize Bangla phonemes using the Euclidian distance measure. Rahman, Hossain, Das,

Islam and Ali (2003) presented a continuous Bangla speech segmentation system using ANN where reflection coefficient and autocorrelations were used as features. They applied Fourier transform based spectral analysis to generate the feature vectors from each isolated words. Authors (Islam, Sohail, Sadid, & Mottalib, 2005) presented a Bangla automated speech recognition (ASR) system that employed a three layer back propagation Neural Network as the classifier. In a research paper Hasnat, Jabir, and Mumit presented an HMM based approach in recognizing both isolated and continuous Bangla speech recognition using HMM models (2007).

Thus, the final challenge for any human-computer interface is the user adaptability and which is what our research stresses on. Jeon, Lee and Bien (2011) stressed on this property of intelligent systems by stating that, the intra-person variation problem in any paradigm of human-computer interaction can be tackled by using fuzzy logic owing to its robustness property against uncertainty and ambiguity. This certainly put solid foundation on the justification of the application of fuzzy logic over HMM techniques in speech recognition.

Fuzzy logic is being used in human commands in the 21st century through voice and gestures analysis as well. Yang, Seung-Eun, Park and Zeungnam(2012) proposed a fuzzy garbage model to provide a variable reference value to determine whether a person's gesture is the command gesture or not .

These significantly modern research works elucidate the precedence of fuzzy logic over HMM based classifiers in human-computer interfacing and among all such interfacing, speech has always held an uniquely dominant position.

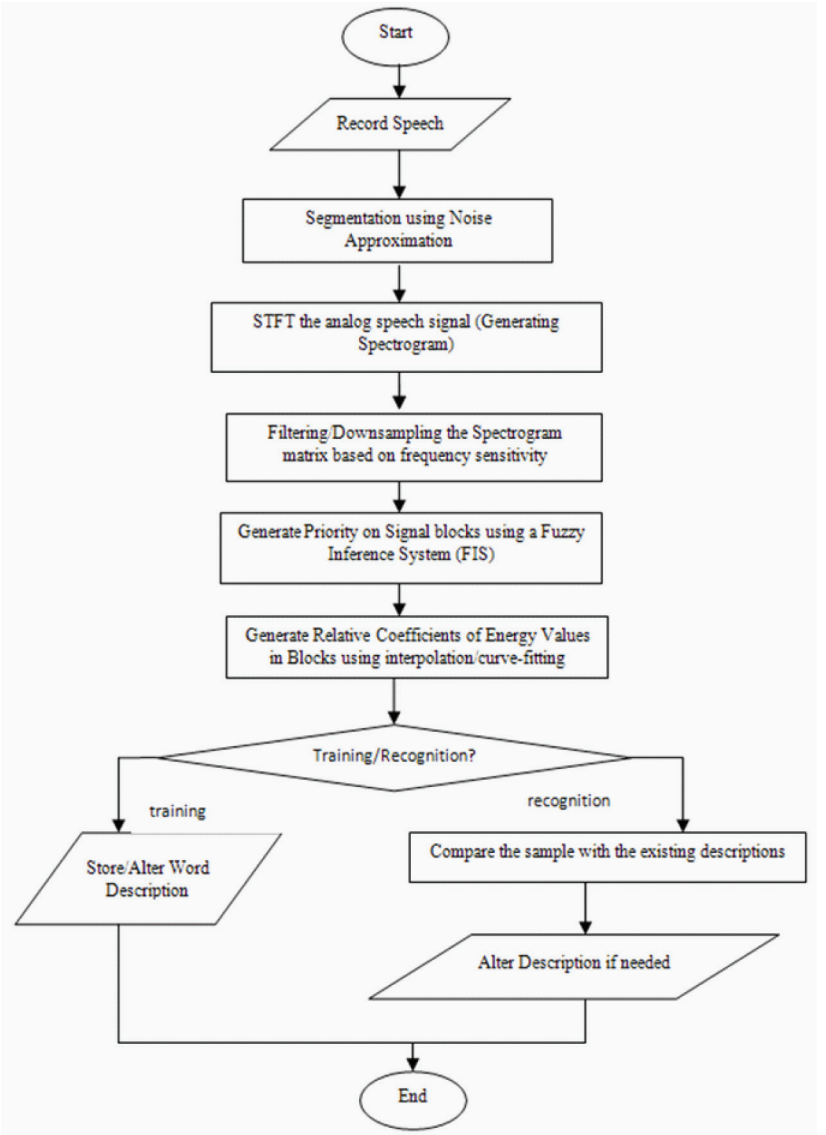
3. METHODOLOGY

The main goal in this research is to create a platform that can translate Bengali speech to Bengali text using two distinct approaches. First one is by generating spectrograms and recognizing speech by using fuzzy logic. The second one is ANN based system that relies on generating MFCC as features for ANN. We compare these two techniques with a marketplace standard HMM based system for accuracy and runtime.

The steps through which we have achieved our goals are vividly illustrated in the flowcharts in Figure 1 and Figure 2, showing the steps for the FIS and ANN approaches respectively.

Since we have approached this problem from two different perspectives (spectral analysis with fuzzy logic and cepstral analysis with ANN), the methodologies become divided into two parts. Since most of the pre-processing (the earlier steps of the methodology) are common to each other, we shall discuss the methodology for the overlapping procedures in the general subsections but for system-dependent methodologies (methods that are different for our FIS and ANN), we have explicitly mentioned them in the titles of the subsections in the Methodology section. Subsections under the Methodology section that do not explicitly mention if it is for FIS or ANN should be regarded as overlapping for both the systems.

Figure 1. A Flowchart showing the methodology of the Fuzzy Logic based system

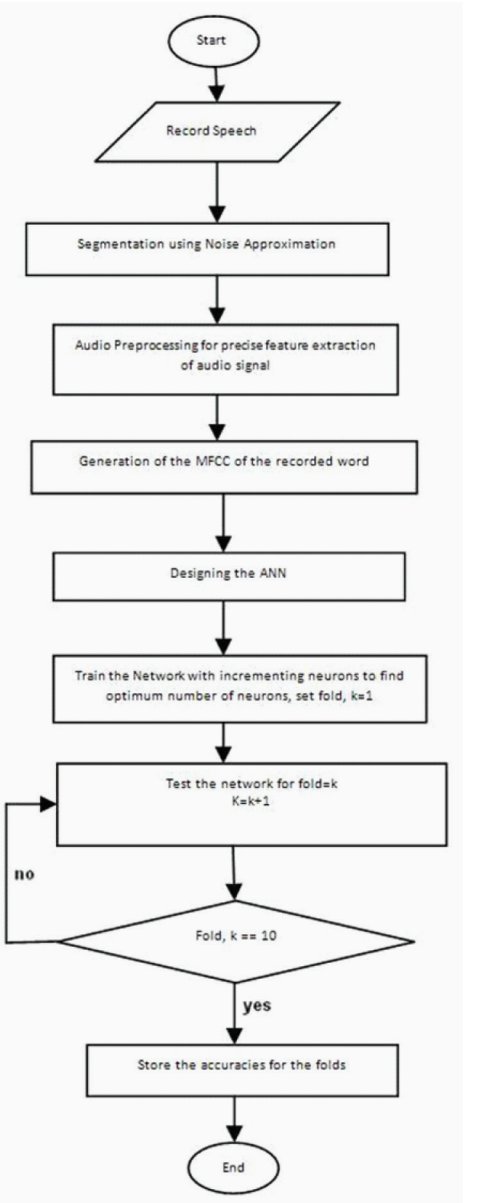


In both our approaches, first we need to train the computer-systems (both FIS and ANN) with original or correct form of utterance of words. Our strategy is divided into two major phases: Learning/Training phase and Recognition phase. Each of the phases consists of multiple steps. Even though the phases are named differently, some of the steps overlap with each other.

3.1. Recording Speech

The first step is self explanatory. Firstly, we get input speech in our systems. Our systems could work on previously recorded "WAV" files or record speech in real time. All sounds that we have used and recorded have the following specifications:

Figure 2. A Flowchart showing the methodology of the ANN based system



- **Bit-Depth:** 8 bit (7 KB/sec)
- **Sampling-Rate:** 8.000 KHz

As the sampling rate is 8 KHz we represent "time parameter" as "sec/8000" in the graphs.

- **Channel:** Mono (since we recognize 'speech', the choice of dual-channel/stereo is meaningless since both channels would give the same signal.)

The recorded word list (classes) and sample size of the dataset (including categorization and number of speakers) of the recorded speech are elaborately discussed in Section 4.

3.2. Segmentation Using Noise Approximation

Before analyzing further, it is imperative that data speech and the descriptions stored in our database need to be phased in such a way that one can be superimposed onto the other. Simply put, if one speaker starts to speak a word after two seconds whereas the description in our database/dictionary starts the data instantly, then the two descriptions will not superimpose properly. Therefore, segmentation needs to be done in a way such that both of the descriptions start from the same point of time. This has been implemented with the following way: when a speaker speaks a word, the system will seek for the level where the amplitude of the signal is greater than 0.2 (relative threshold amplitude that we approximate based on noise of surroundings). A visual representation of a raw and a segmented word is presented in Figure 3 and Figure 4 respectively.

It should be mentioned that, in the figures we use the unit (dB/dB) to relate the "relative amplitude". The term relative amplitude that we use in this research is the raw value of the energy level and the relation of the "relative amplitude" with the conventional intensity unit - decibel (dB) is expressed by the following Equation:

$$y = 10log_{10}\left(x\right) \tag{1}$$

where, x = the relative amplitude

y = amplitude in decibel

Our system considers the signals that are greater than the threshold value. It will ends when the level goes below the threshold level of noise. Thus we find the interval between which the speech exists and create descriptions and compare with the database, based on that segment of the word.

Figure 3. A raw (unsegmented) audio signal for the word "Bangladesh"

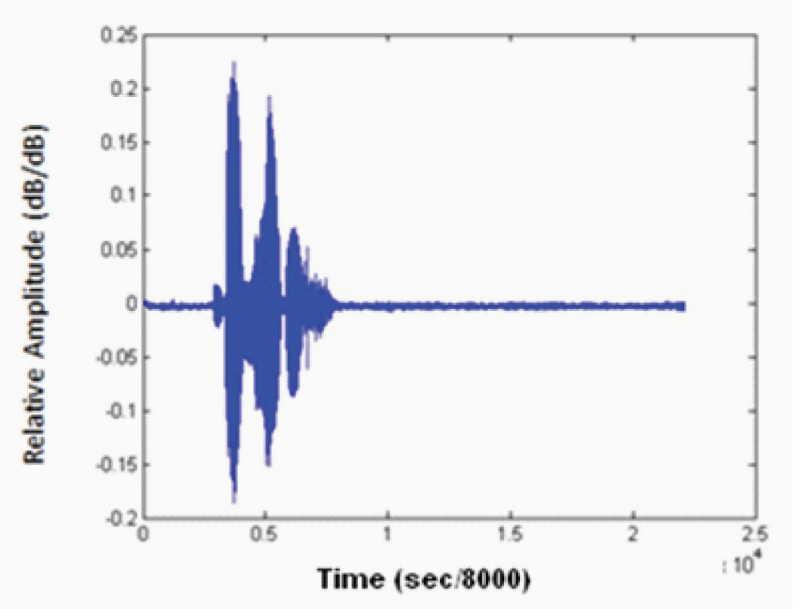
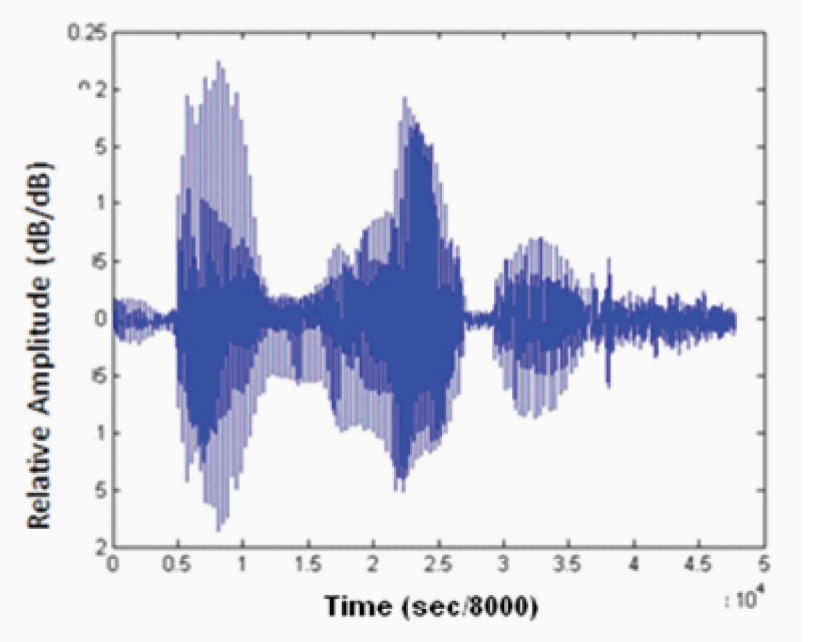


Figure 4. A segmented audio signal for the word "Bangladesh"



3.3. Audio Pre-Processing for Precise Feature Extraction of Audio Signal

From the characteristic of the speech signal, we know that the envelope of the power spectrum of the speech signal shows a 6dB per octave decrease with increasing frequency (Plannerer, 2005). To counter this, we used a pre-emphasis filter as follows:

$$X\left(t\right) = X\left(t\right) - 0.97 * X\left(t - 1\right) \quad (2)$$

Here, X(t) is a digitized sample of the raw recorded voiced signal which is a function of time denoted as t. The time units are segmented into 40 equal windows for the fuzzy based system and we indirectly calculate the t based on Fast Fourier Transform (FFT) indices for the ANN based system (from the generated MFCC).

3.4. Short-time Fourier Transform (STFT) the Speech Data/ Generating Spectrogram

Generation of the spectrogram is one of the core elements of our systems (especially the Fuzzy logic based system). The STFT is a Fourier-related transform that is simplified but often a useful model for determining the frequency and phase content of local sections of a signal as it changes over time.

STFT is defined in Equation (3).

$$\begin{aligned} &STFT\left(x\left(n\right)\right)(m,\omega) \equiv X(m,\omega) \\ &= DTFT\left(x\left(n-m\right)w\left(n\right)\right) \\ &= \sum_{\substack{n=-\infty \\ R-1}}^{\infty}\left\{x\left(n-m\right)w\left(n\right)e^{-\left(i\omega n\right)}\right\} \\ &= \sum_{n=0}(x\left(n-m\right)w\left(n\right)e^{-\left(i\omega n\right)}) \end{aligned} \quad (3)$$

where, DTFT is Discrete-time Fourier transform,

m is the magnitude and ω is the natural frequency of the signal and the signal here is *x(n)* on which the hamming window *w(n)* is applied for the transform. Moreover, *R* is the 'number' of windows accounted for in the positive side; hence, the range of *n* becomes *[0, R-1]*.

The spectrogram of the signal is the graphical display of the magnitude of the STFT, *|X(ω,m)|* which is used in speech processing. The STFT of a signal is invertible that means that we can recreate the sound from the spectrogram using inverse STFT (Smith, 2007). Now let us take a look at Equation (4), where the original Fourier Transform Equation for computing the STFT is given:

$$F\left(k\right) = \sum_{N-1}^{n=0} f(nT) e^{-j\frac{2\pi}{N}nk} \tag{4}$$

where, *f(nT)* corresponds to equally spaced samples of analog time function *f(t)*. But when the samples of the analog function *f(t)* are played through an analog filter, then the frequency response *H(ω)* will be:

$$H\left(\omega\right) = \frac{\sin\frac{NT}{2}(\omega - \frac{2\pi \mathrm{k}}{\mathrm{NT}})}{(\omega - \frac{2\mathrm{k}}{\mathrm{NT}})} \tag{5}$$

Now for determining a running spectrogram and providing flexibility in terms of the filter characteristic, the function used was:

$$F_{t}\left(k\right) = \sum_{N-1}^{n=0} w\left(nT\right) f\left(nT + rMT\right) e^{-j\frac{2\pi}{N}nk} \tag{6}$$

which includes *w(nT),* a new Hamming window for providing a better spectral characteristic (Smith, 2007).

The sample result from this model (in MATLAB) is shown in Figure 5:

In Figure 5, the spectrogram shows frequency on the horizontal axis, with the lowest frequencies at left, and the highest at the right. The colors on the surface represent the amplitude of the frequencies within every horizontal band. Here, the darker the colors i.e. in the "red" family the higher the amplitude/energy-level; except "blue" since blue means 0 as a convention in MATLAB. However according to MATLAB conventions "green" represents energy levels closer to zero.

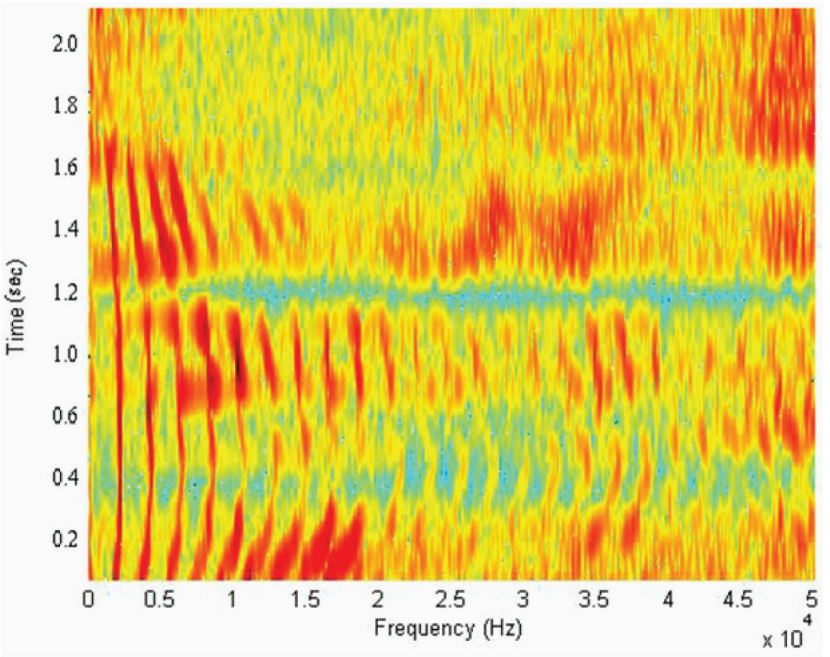
Below we present the exact parameters of the MATLAB function "spectrogram" that we use to generate the spectrograms. The general format of the function is: **spectrogram**(data, hamming window, overlapping-rate, length of FFT, sampling frequency) where the "data" is the raw audio-data. The hamming window, *w(nT)* in Equation (6) is set to 1024 (which is a large amount to achieve high precision). The next parameter is the "overlapping" rate. In Equation 4 - the original Equation for generating a basic spectrogram is shown but in MATLAB we can overlap segments. We chose 1000 overlapping segments that produce 50% overlapping in the segments (and these segments are the core points/pixels of colors in the spectrogram). The next parameter is the "length" of FFT i.e. the *f(nT)* of Equation (4) and it reflects the precision of the division of frequencies. For high accuracy, we chose 1024. Finally for sampling frequency we selected 10^5 since selecting more than this does not produce higher accuracy.

3.5. Filtering/Downsampling the Spectrogram Matrix Based on Frequency Sensitivity (for our Fuzzy Logic Based Bangla Speech Recognition System)

Since the energy levels of a general Bangla word (based on average length) returns a matrix of size 300 x 1400 (based on length of the sound and the parameters of the spectrogram), it is impractical to work with the magnitude of so many values when we want to prioritize 'ambiguity' in our FIS. Thus, we have modeled our fuzzy system to downsample the spectrogram.

However, by using the word "downsampling" in our research, we do not refer

Figure 5. Spectrogram for the word "Bangladesh"



to downsampling of frequency or time rather we refer to reduction of the dimensions of the large matrix corresponding to energy levels to a smaller and manageable dimension which results in better approximation.

We have divided the frequency domain in 30 equal windows and divided the time domain into 40 equal windows. We have used an algorithm such that this segmentation/creating chunks from a large matrix take place in one step. The algorithm works as follows:

Step 1: The frequency domain is divided into 30 equal parts. Since frequency domain in our system always ends in ~ 5KHz, each window gives us a 167 Hz window. Let this value be *x*.

Step 2: Since the time domain is variable as we need to accommodate words of variable lengths. However, we get the highest value as the end timestamp and divide the time domain into 40 equal parts. Let each window be *y*.

Step 3: For an '*x* by *y*' window, we first take the mean of every row/frequency bar (assuming frequency is horizontal) and we select the maximum of the means and we associate that value to that particular chunk.

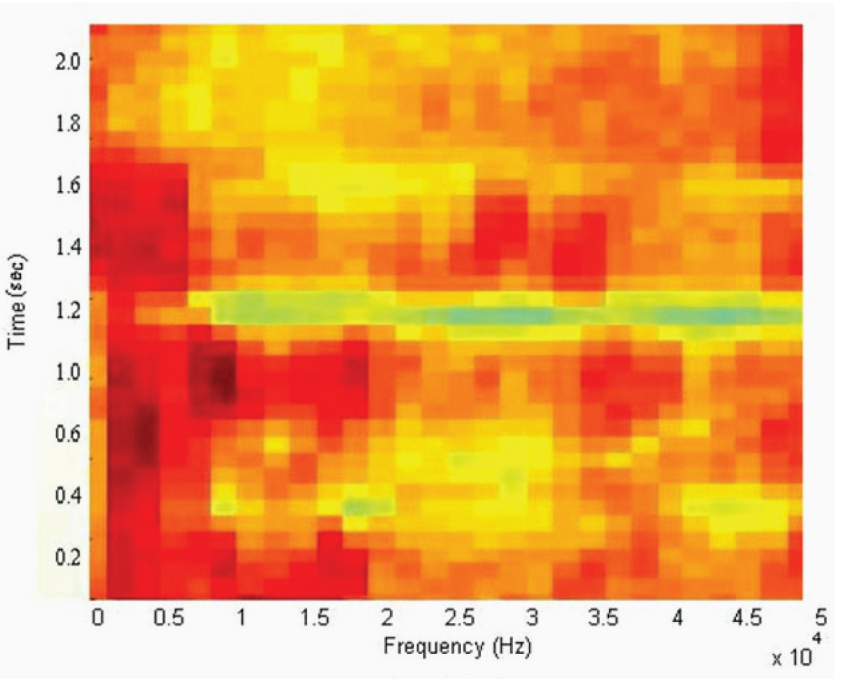
$$\text{value of a single segment} = \bigvee_{i=130} \bigvee_{j=140} mean(S_{ij})$$

Figure 6 illustrates the downsampled spectrogram of the same word shown in Figure 5.

Step 4: We continue step 3 for the whole spectrogram which gives us a matrix of size 30 x 40 where 30 windows are allocated for frequency and 40 windows for time and the values inside the matrix are determined by step 3 of the algorithm.

Since we know that the human ear is more sensitive to lower frequencies and vice versa for higher frequencies, it would be wiser to divide the frequency domain into exponentially growing windows but instead of putting emphasize

Figure 6. Spectrogram for the word "Bangladesh" (downsampled to a 30 x 40 matrix)



on the lower frequencies, we decided to put more importance on the overall evaluation on lower frequencies using our FIS to prioritize the overall "ambiguity" which is the essence of fuzzy logic.

3.6. Storing and/or Altering Word Descriptions in our FIS (for the Fuzzy Logic Based System)

By downsampling the word description matrix (for each word) is reduced to size 30 x 40. For each word, these matrices are stored in our database along with a 'Bengali' string that represents the word speech in text.

3.7. Recognition/ Comparison in our Fuzzy Logic Based Bangla Speech Recognition System

The recognition phase, i.e., the comparison between a recorded sound and a trained definition of a word (in the system) is the foundation of our fuzzy logic based system. Before providing details of FIS we need to mention the following facts about our FIS:

- We have 2 matrices, one for the sample (the word to be recognized) and the other is the target (the description of a word with which it will be compared).
- The similarity of those two will be determined by the closeness of their energy values.
- The similarities will be prioritized by the frequencies.

Next, we present the details of the comparison step:

1. *The Comparison FIS membership functions*: In our FIS, we have 3 inputs e.g. frequency, target and sample. Based on the inputs, the FIS will evaluate the similarity of 2 segments and give us a result ranging in the range [0, 10] here; 10 being the perfect match and 0 meaning no match.

The membership values are illustrated in Figures 7, Figure 8 and Figure 9 that corresponds to Equations (7), (8) and (9). In Figure 7, the *x* axis represents 30 equal and increasing segments of frequencies and the *y* axis represents the degree of membership. Illustrating Figure 8 and Figure 9, it is necessary to mention that all the elements of the downsampled (subsection 3.4) spectrogram – the 30 x 40 matrix of a word description are normalized (ranging from 0 to 1). And such energy values, derived from the STFT powered Spectrogram are represented in the *x* axis of both Figure 8 and Figure 9. In Figure 8, the *x* axis represents the energy level (every element of the 30 x 40 matrix of a word description) of a particular word in our database (which we are calling "sample"). On the other hand, the *x* axis of Figure 9 represents the normalized energy level of a word spoken by a user (which we are calling "target") which will be compared to "sample" as explained above. These energy levels are also merely the elements of the 30 x 40 matrix generated through "downsampling" (subsection 3.5) from the original spectrogram/STFT (subsection 3.4) of the voiced data. In both Figure 8 and Figure 9, the *y* axis represents the degree of membership.

$$\mu_{frequency}(x) = \begin{cases} -\frac{x}{11}+1, 0 \le x \le 11 (x \ is \ low) \\ \frac{x-6}{9}, 6 \le x \le 15 (x \ is \ medium) \\ \frac{-(x-22)}{7}, 15 \le x \le 22 (x \ is \ medium) \\ \frac{-(x-30)}{14}+1, 16 \le x \le 30 (x \ is \ high) \end{cases} \quad (7)$$

$$\mu_{sample}(x) = \begin{cases} \frac{1}{1+0.4(x-0)^3}, 0 \le x \le 0.5 (x \ is \ small) \\ \frac{1}{1+0.4(x-\frac{1}{2})^{2.5}}, 0 \le x \le 1 (x \ is \ medium) \\ \frac{1}{1+0.4(x-1)^3}, 0.5 \le x \le 1 (x \ is \ large) \end{cases} \quad (8)$$

$$\mu_{target}(x) = \begin{cases} \frac{1}{1+0.4(x-0)^3}, 0 \le x \le 0.5 (x \ is \ small) \\ \frac{1}{1+0.4(x-\frac{1}{2})^{2.5}}, 0 \le x \le 1 (x \ is \ medium) \\ \frac{1}{1+0.4(x-1)^3}, 0.5 \le x \le 1 (x \ is \ large) \end{cases} \quad (9)$$

The membership functions used in our systems have been based on our sole understanding of the recognition of speech. However the membership function for "frequency" (Equation 7 and Figure 7) has been in accordance with the conventional model of MEL spaced filterbanks (IIFP, 2010) even though it has been modified as illustrated in Figure 7.

$$\mu_{similarity}(x) = \begin{cases} -\frac{x-3}{3}, 0 \le x \le 3 (x \ is \ very \ low) \\ \frac{x-2}{2}+1, 0 \le x \le 2 (x \ is \ low) \\ \frac{-(x-4)}{2}, 2 \le x \le 4 (x \ is \ low) \\ \frac{(x-4)}{2}+1, 2 \le x \le 4 (x \ is \ medium) \\ -\frac{(x-6)}{2}, 4 \le x \le 6 (x \ is \ medium) \\ \frac{(x-6)}{2}+1, 4 \le x \le 6 (x \ is \ high) \\ -\frac{(x-8)}{2}, 6 \le x \le 8 (x \ is \ high) \\ \frac{(x-8)}{2}+1, 6 \le x \le 8 (x \ is \ very \ high) \\ -\frac{(x-10)}{2}, 8 \le x \le 10 (x \ is \ very \ high) \\ \frac{(x-10)}{2}+1, 8 \le x \le 10 (x \ is \ perfect) \end{cases} \quad (10)$$

All the other membership functions (sample, target and similarity) were derived from the visual representation of the membership function shapes (modeled by ourselves using MATLAB) based on logical perception (with a Gaussian inclination).

Figure 7. Membership function for frequency corresponding to Equation (7)

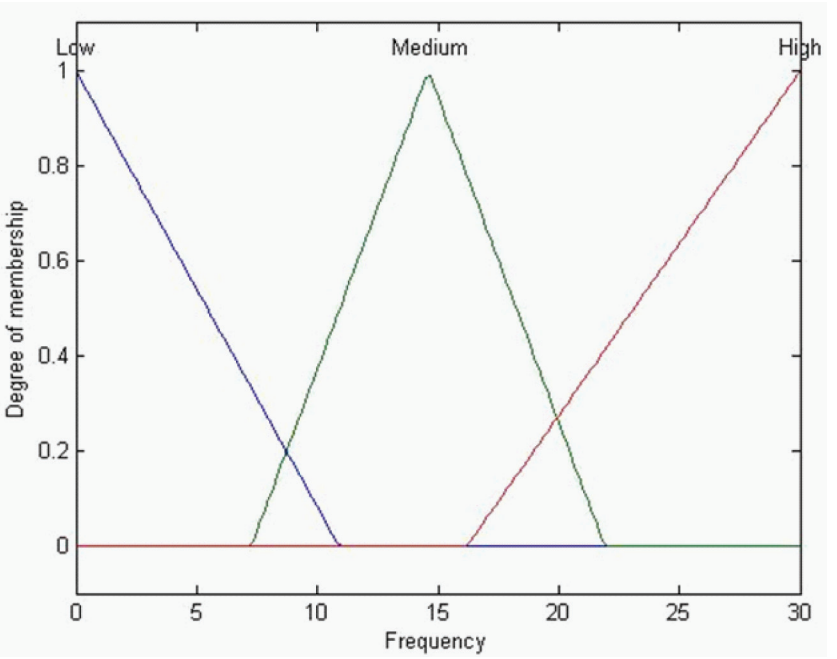


Figure 8. Membership function for sample (energy level) corresponding to Equation (8)

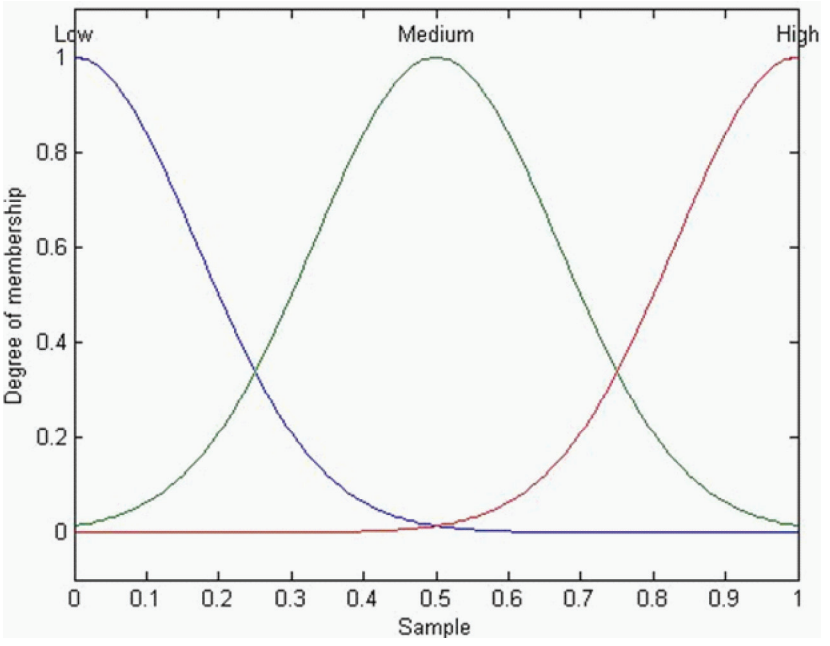
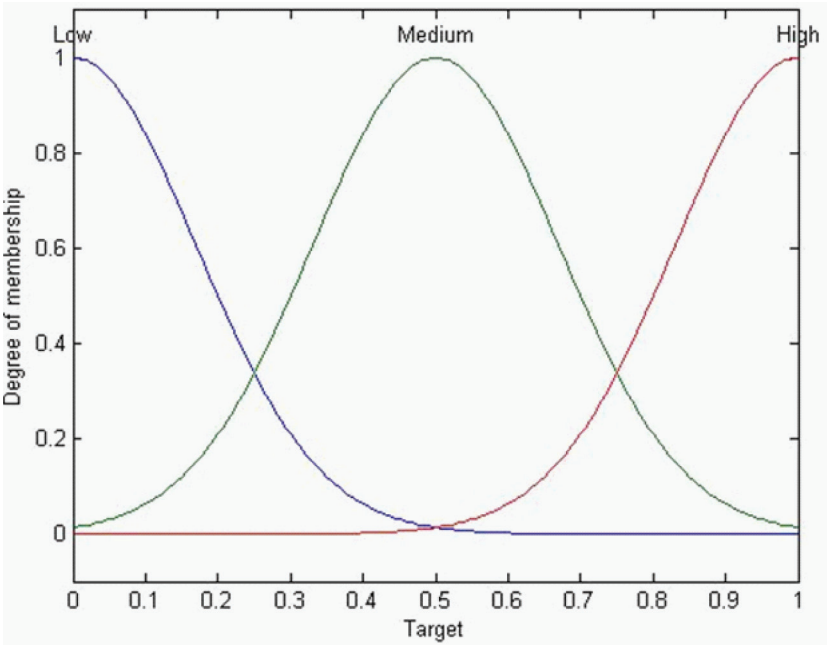


Figure 9. Membership function for target (energy level) corresponding to Equation (9)



2. The fuzzy if-then rules: As we have 3 inputs, we slice the three dimensional table into three two dimensional tables (Table 1-3). Because of having 3 input variables, we use 27 fuzzy propositions (fuzzy if/then rules) to model our system. Also, two surface plots evaluating the model are presented in Figure 11 and Figure 12.

The elaborations of the terms in tables are given below:

VL = Very Low
L = Low
M = Medium
H = High
VH = Very High
P = Perfect

From the rules it can be inferred that the lower frequencies has been given higher priority when evaluating the rules.

To get the complete view of the evaluation of the FIS, Figure 11 and Figure 12 has been generated using MATLAB. These two figures show the evaluation of the aggregate of all the 27 fuzzy propositions (if/then rules presented in Table 1, 2 and 3) and the membership functions shown in Figure 7-9 as surface plots. Since we have 3 input variables (Frequency, Sample and Target) and 1 output variable (Similarity), the total number of variables is 4 which cannot be accommodated in a single figure (as four dimensions cannot be represented in a 3 dimensional form). Thus, we are using 2 figures to illustrate the overall evaluation of the FIS.

In Figure 11, the frequency (ranging from 0 to 30 as defined in the membership function in Figure 7) is represented in the *x* axis. Here the variable "Target" is the word description's energy values (in the form of a 30 x 40 matrix) that the speaker has spoken which the system identifies. Since it is normalized it is ranging from 0 to 1 (Figure 9 shows the membership function). Finally the similarity (ranging from 0 to 10 – based on our modelling as represented in Figure 10) is represented in the Z axis as output.

Figure 10. Membership function for output – Similarity, corresponding to Equation (10)

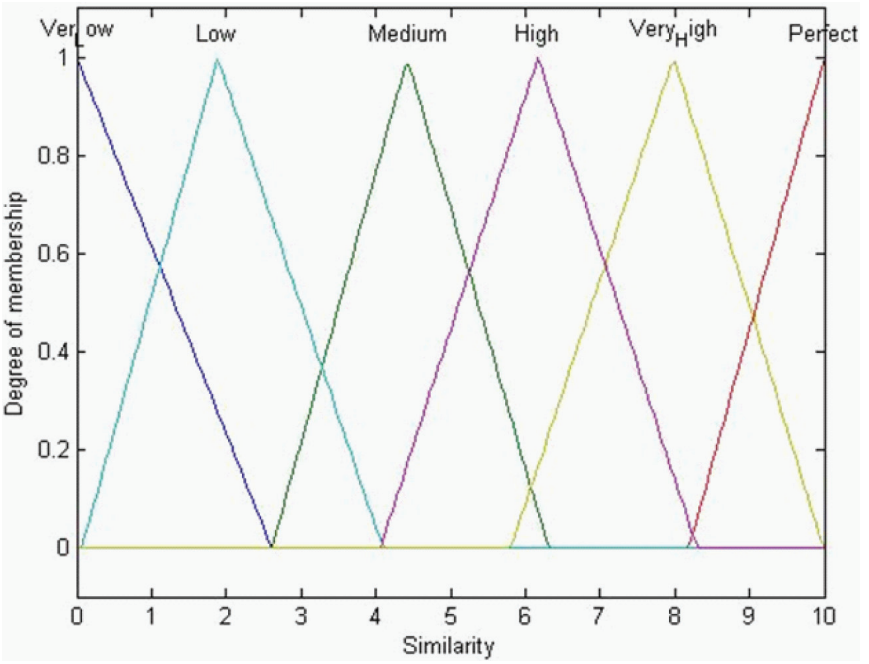


Figure 11. Surface illustrating the evaluation of the FIS representing Frequency (X axis), Target (Y axis) and similarity (Z axis)

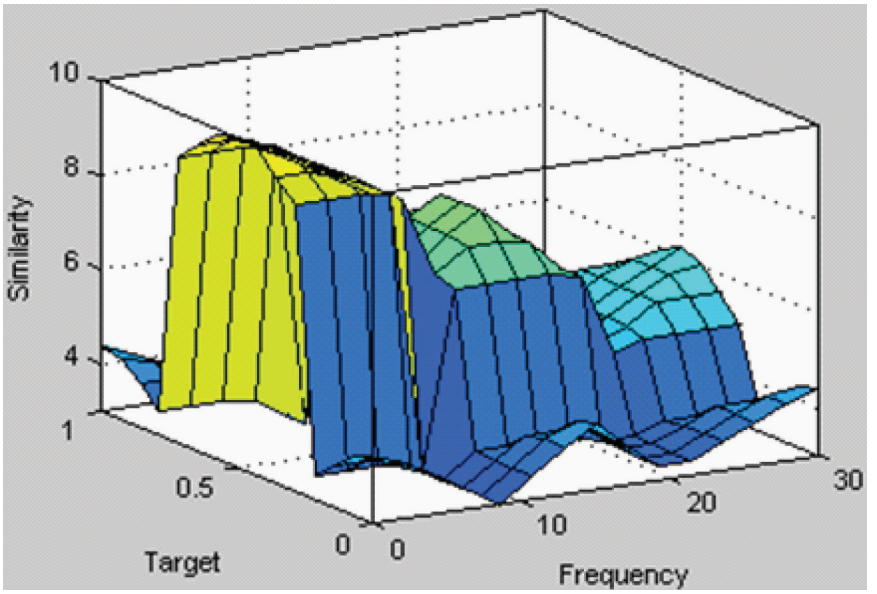


Figure 12. Surface illustrating the evaluation of the FIS representing Frequency (X axis), Sample (Y axis) and similarity (Z axis)

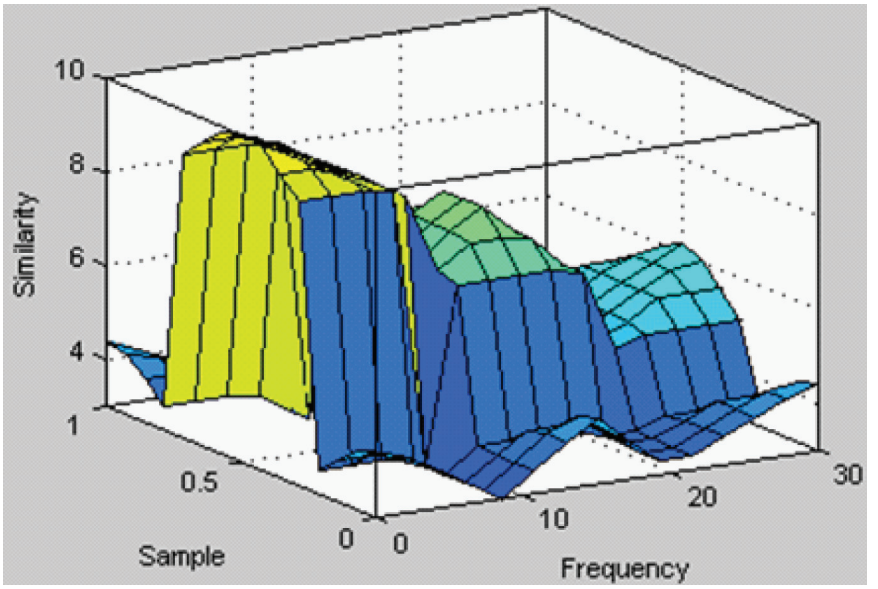


Table 1. Fuzzy rules when frequency is low

		Sample		
		Lsample	**Msample**	**Hsample**
Target	Ltarget	P	M	L
	Mtarget	M	P	M
	Htarget	L	M	P

Table 2. Fuzzy rules when frequency is medium

		Sample		
		Lsample	**Msample**	**Hsample**
Target	Ltarget	VH	M	L
	Mtarget	M	VH	M
	Htarget	VL	M	VH

Table 3. Fuzzy rules when frequency is high

	Sample			
		Lsample	**Msample**	**Hsample**
Target	Ltarget	H	M	L
	Mtarget	M	H	M
	Htarget	L	M	H

But Figure 11 cannot independently shed light on the overall evaluation of the FIS. For that we need to look at Figure 12 as well. Here the X axis and Z axis are same as Figure 11 (frequency and similarity, respectively). However, the Y axis is now "Sample" which is the word description (energy values of the 30 x 40 matrix) to which the "Target" matrix is compared. By "Target" we refer to a particular word description (for a particular comparison) in our data-set in form of a 30 x 40 matrix. Since the energy values are normalized, they range from 0 to 1 (Figure 8 and Figure 9 show the membership functions of "Sample" and "Target" respectively).

It is important to note that the "Target" word description changes to the next word in the data-set every time a particular comparison of a word in the dataset is computed against a "Sample" (user's uttered speech). Simply put, a linear search is done in an array of word descriptions (the array of "Targets") searching for the closest match to "Sample."

Now if we look at the surface plots of Figure 11 and Figure 12, we notice that both the surfaces are similar. It is because both "Sample" and "Target" are normalized and they are reflected on the similarity fuzzy propositions (Table 1-3). Moreover, for achieving accurate similarity between "Sample" and a specific "Target", the membership functions for both (Equation (8) and Equation (9) respectively) are modeled as identical. In other words, if we picture the two figures together, then the similarity will reach towards 10 (which is illustrated in Figure 10 as a membership function plot) if "Sample" and "Target" has the same or close values (based on the fuzzy if/then rules).

It should be also noted that Figure 11 and Figure 12 conveys the fact that, lower frequencies are getting higher priorities in the FIS than higher frequencies. If we examine and go along the X axis (which represents "Frequency") then we see that the height of the surface gradually decreases (in both Figures 11 and 12). This tells us that as we move to higher frequencies from lower ones, the similarity found between "Sample" and "Target" are gradually given less weight in "Similarity" which coincides to the fuzzy if/then rules represented in Tables 1-3. For instance, if an energy value of a particular segment (one particular value in the 30 x 40 matrix of Sample) matches completely with the energy value of the segment of the same location of the "Target" (which is another 30 x 40 matrix) then the similarity will not always be 10. According to Figure 11 and Figure 12, if this match is found when relative frequency value is one (every relative frequency value corresponds to 167 Hz rising up to 5 KHz as discussed in Subsection 3.5) then the similarity is given as ~ 9.9 on a scale of 10 (paying more priority to lower frequencies), however if this same matching takes place where frequency value is 25 (which corresponds to 25 x 167Hz = 4175 Hz as mentioned in Subsection 3.5) then the similarity value (defuzzified) is ~ 6.5 (on a scale of 10).

3.8. Computation of Signal Energy of Speech (Based on Sampling rather than Spectrogram) for the ANN Based System

The original signal energy of the raw audio is used as a feature for recognition for the ANN based system. It is computed with Equation (11).

$$E = \sum_{k=1}^{n} X^2(k) \tag{11}$$

Here,

E: signal energy of raw audio recording of a word. Even though the ideal unit of energy is Joules but in practice of speech and signal analysis we express it more commonly in terms of amplitude and frequency. Here, amplitude is calculated in dB and frequency in Hz.

X(k): k^{th} digitized sample of the recorded signal. It is to be noted that here, the term "sample" does not refer to a particular entity or word in the dataset, rather it refers to the audio signal of a particular frequency inside the range that we have chosen as elaborated in subsection 3.5. Thus, here k refers to the different frequencies in which we have done sampling of the analog voiced signal to its digitized representation.

n: number of samples of frequencies based on which we have sampled the analog voiced signals.

Even though the spectrogram (discussed in subsection 3.4) returns energy values, we computed the signal energy values based on a more generalized formula stated in Equation 11 to create autonomy between our ANN based system and our Fuzzy logic based system, even though both will concur similar responses but in different forms (i.e. the Fuzzy logic based system did not use the mel scale explicitly; rather it had used it implicitly by defining a membership function in the fuzzy system model to meet its concordance). Another justification is the conformity to MFCC (elaborated in the next subsection – subsection 3.9).

3.9. MFCC Generation for Feature Extraction for the ANN Based System (Cepstral Analysis)

MFCC is the most widely used feature for speech recognition. To compute MFCC from the obtained spectrogram, we map the powers of the frequencies from the spectrogram for each time window in Mel Scale by using triangular overlapping windows. As there is no 'single' standard Equation for the conversion of the unit of frequency from Hertz to Mel, the most popular one (and the one we used) is used which is Equation (12):

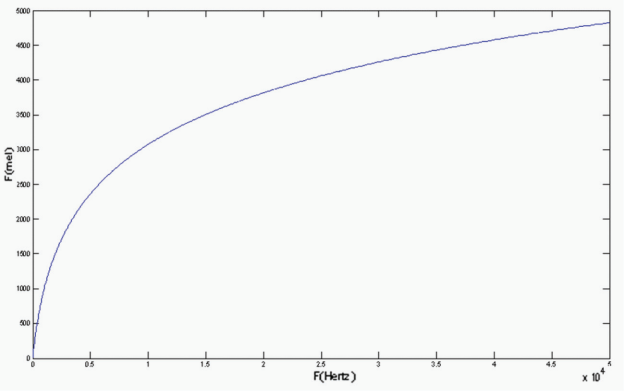
$$m = 2595 log_{10}(1 + \frac{f}{100}) \tag{12}$$

where, m is the mel-scale frequency and f refers to frequency in Hertz (Traux, 1999).

It is evident from Figure 13 that MEL scale is much more sensitive to changes in the lower frequencies of the Hertz scale which coincides with the hearing perception of the human ear.

The energy spectrum and the mel-frequency cepstrum (MFC) differs in the sense that in the MFC, there is equal spacing of the frequency bands in mel scale, which emulates the human auditory system's response more closely than the linearly-spaced frequency bands used in the regular spectrum. This frequency deformation allowed us to represent the sound signal more accurately as heard from the perception of the human ear (Molau, Pitz, Schlüter, & Ney, 2001). To obtain the power spectrum in mel scale we compute a filter matrix from the indices of FFT. Then we filter the power spectrum with the obtained filter matrix to obtain the values in mel scale. After that, we apply 'log' to this spectrum to separate unwanted ripples from the audio signal. Lastly, the ripples are removed from the spectrum by performing a Discrete Cosine Transform (DCT) and thus obtaining

Figure 13. Graphical relationship between Hz scale and MEL scale



the mel cepstrum. This process is also known as cepstral smoothing.

The discipline of this modified representation of the frequency bands is referred to as Cepstral analysis and our core feature for the ANN based system is based on this particular form of signal analysis.

In our ANN based system, we used 60 Mel values that are equally spaced in the Mel Scale (Figure 14), and took the first 25 coefficients (ignoring the DC Offset) to emulate a low pass filtering process (also known as liftering) done in human ear (Figure 16). Plannerer proposed the selection of the first 13 coefficients out of 50 generated MFCC (2005) and on the other hand a standardized MFCC algorithm to be used in mobile phones was introduced by The European Telecommunications Standards Institute where first 40 out of 80 coefficients were selected for digit recognition (Traux, 1999). Through our analyses we selected the first 25 coefficients out of the computed 60 MFCCs to achieve a high accuracy for Bangla words and also considering a resource trade-off.

In Figure 15, we see that the vertical axis ranging from 1 to 25 as we justified in this subsection and the horizontal axis represents the centre of the time samples where FFT was applied, making the unit - time/FFT_Window_size, thus being sec/8000 (as we selected window size to be 1/8000 seconds since we wanted it to coincide to our original sampling rate of audio signal which was 8 KHz). The colors follow the same convention as illustrated in Figure 5.

This data is then simplified by reducing the time domain into 10 partitions applying equally sized windows and taking the maximum value of the spectrum to represent that window. This novel approach gives us the most dynamic responses of the audio signal leaving off the lower ones.

To trace the dynamic change in the cepstrum, we compute the derivative and second derivative of the MFCC. The derivative is computed from the difference between a cepstral coefficient value and the one prior to it in the time domain. The second derivative is obtained applying the same process on the derivative. Then the derivatives are also undersampled in the horizontal axis into 10 values using the same process followed for the cepstrum.

Thus we see that, unlike the 30 x 40 undersampled matrices for each word description in our FIS, a speech signal description is now represented by 3 x 25 x 10 = 750 features (the 3 corresponds to original MFCC, the derivative of the MFCC and the double derivative, 25 is the number of coefficients selected for the design and 10 is the number of undersampled windows along the time domain. These values along with the signal energy are normalized and used as features for the ANN.

Figure 14. Mel Filter bank matrix required to map the powers of the Fourier Spectrum into Mel Scale

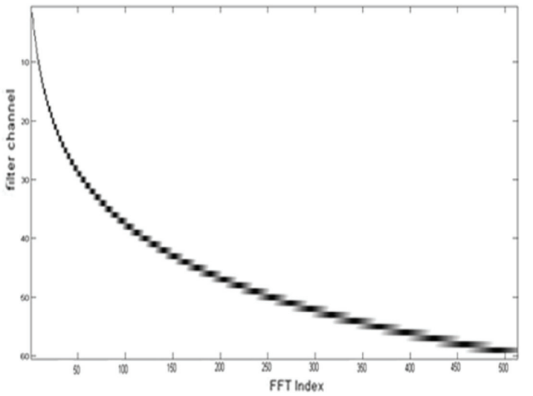
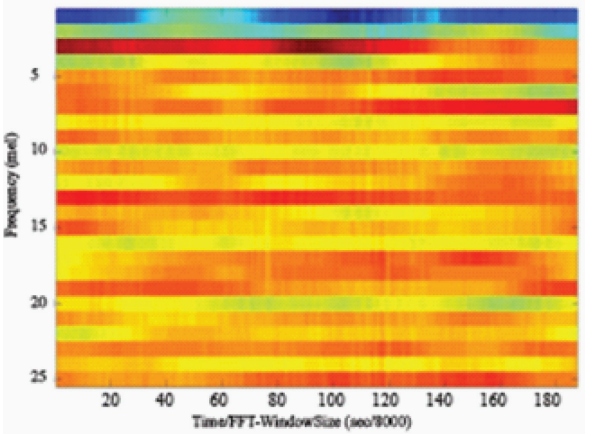


Figure 15. Graphical representation of MFCC of the word "Bangladesh"



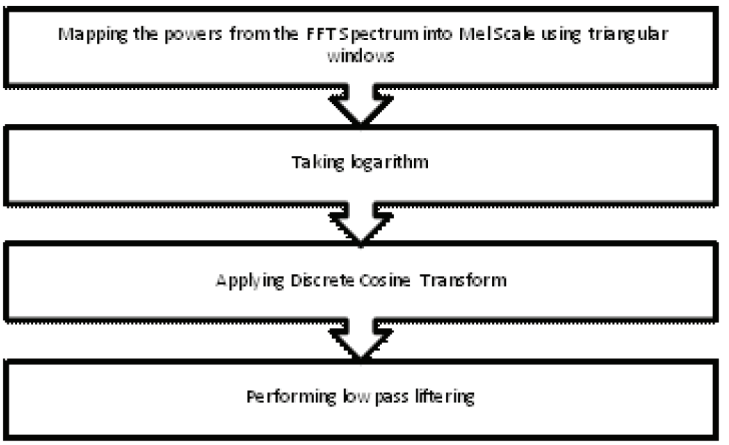
3.10. Designing the ANN (Feed Forward Backpropagation Artificial Neural Network) Based System

Now we shall discuss the design of our ANN based system in three subsections. In subsection 3.10.1, 3.10.2 and 3.10.3 we discuss the number of layers (the original design of our network), the selection of number of neurons in the hidden layer (with justifications) and the selection of our training algorithm respectively.

3.10.1. Design (selection of number of layers) of the Architecture of the ANN Based System

From definition of ANN design, we know that the first layer is the input layer which corresponds to the features (every element of a word description matrix), the output layer corresponding to class information (mapping of every distinct word as a class) and in between the input and output layers there can be one or multiple hidden layers. For coherence and meaningful analysis, we choose to take one hidden layer in between the input and output layers.

Figure 16. Process of obtaining MFCC from Fourier Spectrum



The justification of this selection can be given as follows: since the number of neurons of the input and output layers are fixed or constant based on features and class numbers respectively, we shall be able to parameterize and change the number of neurons of the single hidden layer and test the performance of the network for different numbers of neurons in the hidden layer which consecutively helps us in selecting an optimum number of neurons to design our ANN based Bangla speech recognition system. The details of this selection are discussed in the following subsection (Subsection 3.10.2).

3.10.2. Selection of Number of Neurons in the Hidden Layer of the ANN Based SYSTEM

Conventionally, the first layer or the input layer contains all the features of a word description and the output layer contains numbers of neurons equals to the number of words in the database or dictionary of word descriptions we are using; since we have 50 words (or 'classes' in ANN linguistic jargon) in our database/dictionary, it contains 50 neurons in the output layer.

Therefore, the variable we are concerned with is the number of neurons in the layer in between the input and output layer (the hidden layer). We have trained the network with incrementing number of neurons in the hidden layer (with several training algorithms – discussed in details in the next section) and plotted the response or accuracies in a "number of neurons in the hidden layer vs. accuracy graph". We prioritized the recognition accuracy rather than the training time for gaining better accuracy. All of this information is vividly illustrated in Figure 17.

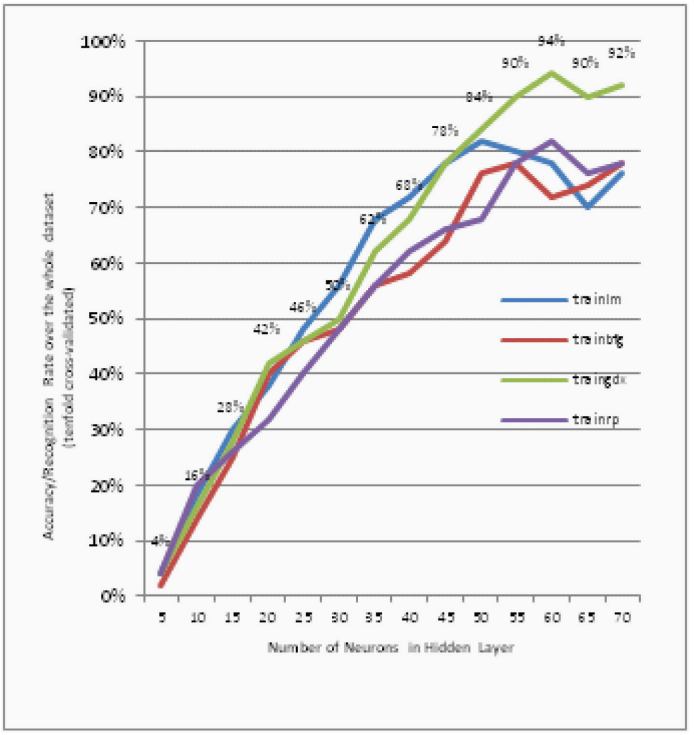
From the figure, we can vividly see that we get the highest accuracy of 94% (after tenfold cross validation – explicitly explained in subsection 3.11.2a) with 60 neurons in the hidden layer. Therefore, we can conclude that our network design is a 750-60-50 network (as per convention input_neurons - hidden_layer_neurons - output_layer_neurons).

Thus, from this analysis, we decided to test each of our testing categories and comparisons using 60 neurons in the hidden layer of our ANN.

3.10.3. Selection Training Algorithm for the ANN based System

Since we had 750 numbers of features (for each word sample) and 50 words and 20 samples of each word (10 samples from a male and 10 from a female which are elaborated in the Section 4 – Result Analysis) in our data dictionary/ database it took considerable time to train our network. Rather than randomly taking any random training algorithm we trained and tested

Figure 17. Number of neurons in hidden layer vs. tenfold cross-validated accuracy (%) over whole dataset of 50 words (using 4 training algorithms)



the performance of our network using 4 most popular training algorithms for feedforward backpropagation networks. They are:

- Trainlm (Levenberg-Marquardt backpropagation)
- Trainbfg (BFGS quasi-Newton backpropagation)
- Traingdx (Gradient descent with momentum and adaptive learning rate backpropagation)
- Trainrp (Resilient backpropagation)

(Mathworks, 2011)

As discussed in the previous subsection, we choose the training algorithm for the distinct subsections of the result analysis (recognition accuracy to be precise) based on Figure 17. We prioritize the recognition over the throughput or training time (Figure 17). Thus we choose to trade-off training time over accuracy since we put more importance on the accuracy of our network than training time. The difference made by the trade-off selection will still be further explained in the training-time subsection of the result analysis 3.11.2e section.

As we can see from Figure 17, we reach the highest accuracy irrespective of the training time (elaborately discussed in subsection 3.8.2e) for the training algorithm "traingdx" (Mathworks, 2011) and thus we choose it for our further analyses. More elaborate justification for this choice is given in subsection 3.11.2e.

3.10.4. Selection of Transfer Functions for the ANN Based System

Finally, we choose the transfer functions for the layers of our ANN based system. For the hidden layer we have chosen 'tansig' as it squashes the feature values in the interval of [-1,1] which

enables optimized training times for training algorithms. Then, for the output layer we have chosen 'purelin' transfer function to preserve the outputs. However, in order to round off and zero out erroneous values we have used an additional competitive transfer function – 'compet' such that it returns only a single "1" as output for a resultant class and all 0's for the rest.

3.11. Training

After setting up the FIS and our ANN based Bangla word recognition system, the next task is one of the most important in the expert systems paradigm, which is 'training'.

3.11.1. Training the Fuzzy Logic Based Recognition System (FIS)

After we feed our fuzzy system with 1000 words (50 Bangla words spoken 10 times each by a male speaker and a female speaker = 50 x 10 x 2 = 1000), we get all word descriptions (in form of a 30 x 40 fuzzy set for each word) for training and the user gets the liberty to alter the description based on his/her particular voice/ tone/stress etc.

If a particular word – spoken by the user is recognized incorrectly, then the system asks for the correct word from the user i.e. the user will type in which word he/she has just spoken if and only if the system fails to recognize the word itself. After getting the inputs (voiced signal – which, in turn will be converted into a 30 x 40 matrix as discussed in Subsection 3.5, and the correct word string) – the system compares this user's input (voiced data that has been converted into a 30 x 40 matrix as explained in subsection 3.5) with the description of our database. Based on the difference of the two, the description stored in our database is altered in accordance with the users speaking. Consequently, the incorrect description will shift towards the user's version of the word.

To clarify, let us consider that a user has spoken a word that the system has recognized incorrectly; suppose, user said the Bangla word "Ek" but the system recognized it as the Bangla word "Aat". In that case the user prompts the system that the match was incorrect and this information is stored in a 'Boolean' variable to designate if the sample was a match or not. If it was not a match i.e. the system recognized the word incorrectly, then the word description of the original word will be altered as follows:

$$E_{level(new)} = \left(E_{level(original)} * original\_weight\right) + \left(E_{level(training)} * training\_weight\right) \quad (13)$$

Here,

$E_{level(new)}$ = the energy level of a particular segment of the spectrogram (every element of the 30 x 40 matrix of the word description and for this example it is the description for the word "Ek")

$E_{level(original)}$ = The energy level that was stored in the database for the word that was spoken (or being trained) by the user (and in this example the word is "Ek").

$E_{level(training)}$ = The energy level that the user had just spoken which was identified incorrectly (and in this example the word is "Ek" which was incorrectly identified as "Aat").

Original_weight = the weight of the energy level of the original word description stored in the database. We chose, Original_weight $= \frac{1}{2}$.

Training_weight = the weight of the energy level of the word spoken by the user.

We chose, Training_weight $= \frac{1}{2}$.

Therefore, we see that the summation of both the weights gives us 1 (and whatever weight we choose for modeling, the summation of the 2 weights has to be 1), and from this we infer that the original word description stored in the database will shift 50% towards the word description of the word that a user has just spoken.

On the other hand, the same system ends up with different word description with different people, making it adaptive to the speaker's

tone. Thus, our system becomes user-adaptive with time. Therefore the verdict can be reached that with time our system develops more and more accuracy for a particular user.

3.11.2. Training the Feedforward Back-propagation Artificial Neural Network Based Recognition System (ANN)

Before we go into the discussion of training the ANN, the concept of a very popular training and validating technique named k-fold cross validation needs to be explained in order to comprehensively describe our training and result analysis section since we have used that standard i.e. we have used ten-fold (k=10) cross validation to train and test our system.

1. **Ten-Fold Cross Validation:** By tenfold cross validation, it means that we had to train the network with 90% of the data and after training we test it with the remaining 10%. After calculating the performance, we had to take "another" or "next" 10% of the data for testing and train the network with the "rest" 90%. Since our dataset was of 1000 Bangla word descriptions, the number of Bangla word descriptions for training data was 900 word descriptions and testing was of 100 word descriptions. Thus, with 10 steps/folds we were done with one comprehensive training (and testing) of the complete data-set. But the data-set was divided into categories to further illustrate the results which are discussed in Section 4. The strength of k-fold cross validation is the fact that training data and testing data will never overlap and thus, it will minimize false positives and will not result in an incorrectly high accuracy of a system.
2. **Mapping Bangla Words to Classes:** Since feedforward backpropagation ANN is a supervised network, we had to map each word to a particular class. Thus, we designated each word to a class to train the network that is explained in the next section. In a nutshell, the first distinct word (we need to be careful to be not mixing up "word/class" with "sample" since there are 20 samples of each class/word as discussed in subsection 3.11.1 and Section 4) was designated as class 1; second word was class 2 and repeating this we get to class 50.
3. **Creation of the Training Matrix:** In order to train the network first we classified the words. Then, the next job is to create a matrix containing word descriptions of the training data (900 words as discussed in subsection 3.11.2a for ten-fold cross validation).

Following the conventions of Neural Network Toolbox of MATLAB, we converted each word description into a single column where each row represents a feature. Therefore for each fold there were 900 columns with 750 features (rows) (refer to subsection 3.9 for generation of features for the ANN). Even though for each fold (among the 10 folds) the training data changed, the dimension of the training data remained the same (750 x 900 features).

4. **Creation of the Target Matrix:** As this is a supervised network, a target matrix had to be generated. There is no "right" or "wrong" way of creating one but it is up to the developers. We designed the target matrix such that every column represents a word/class from the testing data and the "correct" word or class that it represents contains 1 and all the other rows of a column contains 0's.

We designed the matrix as follows in Table 4 so that every next word is a new word and such that it conforms to the "training" data for convenience of matrix creation.

Thus, we can see that the dimension of the target matrix remains the same for all the folds which is

50 x 100 (as there are 50 words in the complete dataset of 1000 samples and each word is uttered 10 times by a male and then a female).

Table 4. The "Target Matrix" (the numbers in "bold" are indices of the matrix)

	1	**2**	**3**	…	**49**	**50**	**51**	**52**	…	**99**	**100**
1	1	0	0		0	0	1	0		0	0
2	0	1	0		0	0	0	1		0	0
3	0	0	1		0	0	0	0		0	0
4	0	0	0		0	0	0	0		0	0
		. . . .					. . . .				. . . .
49	0	0	0		1	0	0	0		1	0
50	0	0	0		0	1	0	0		0	1

The dimensions are 50 x 100 as there are 50 classes or words and in each fold we tested 100 samples (10% of the complete dataset as discussed in subsection 3.11.2a).

5. **Training the Network:** Once we have prepared the training dataset (for each fold) and the target matrix, the system becomes prepared to be trained. Training a network is time consuming (and it often takes several hours) for a large dataset as ours with numerous features (since we did not perform any dimensionality reduction because we wanted to preserve all the attributes of the sound for the ANN system since it is manageable). But it is to be noted that if feature size goes beyond the scope of computation then dimensionality reductions techniques such as PCA or LDA can be applied to reduce it (Chen, 2012). However, we did not need to perform them as our training successfully converged using all the 4 training algorithms we trained it with.

Training times varied significantly based on training algorithms. As mentioned in subsection 3.10.3, we chose the training algorithm which exhibited the most precise accuracy over economy of training time. The comparative illustration of training times of our system in the 4 algorithms we trained it with is illustrated in Figure 18.

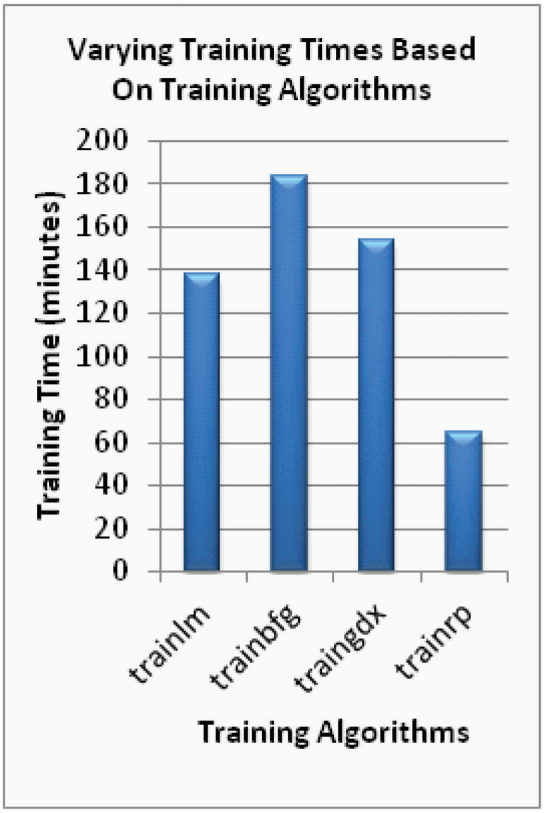
It is imperative to mention the configuration of the computer to train the system as the training times are subject to change based on computer configuration (clock speed and main memory). These training times were generated in a computer using Intel Pentium Core 2 Duo Processor with clock speed of 2.80 GHz using 3 GB RAM.

Here, we can see that choosing "trainrp" would have been the best choice since the training time and consequently the convergence epochs are less. But since we decided to give more priority on accuracy, we have chosen "traingdx" for our primary comparisons in the result analysis. The justification for this selection is given in Figure 17 and in subsection 3.10.3.

4. RESULT ANALYSIS

We have tested our systems by categorizing Bangla words in 3 categories. They are – monosyllabic, bi-syllabic and poly-syllabic. The total number of Bangla words we tested for our systems were 50 (20 of which were monosyllabic, 20 were bi-syllabic and 10 were polysyllabic). Among the 50 words/classes, each had 10 recordings/utterances done by a male and a female speaker (every speaker – one male and one female uttered one word ten times). Thus, the complete dataset was of 1000 samples or word descriptions (50 x 10 x 2). All the tests and results

Figure 18. Training algorithms vs. training times (minutes) graph



followed the ten-fold cross validation system explained in subsection 3.11.2a. Not more than one speaker of a particular gender (one male and one female) was used instead of multiple individuals intentionally, in order to prove the "user adaptation" quality of our systems which was discussed elaborately in subsection section 3.11.1, where we state and claim theoretically that our systems, may it be trained by anyone, if used and further trained by "any" user for a significant amount of time, will adapt to the user's voice and the accuracies are subject to only increase gradually and theoretically, and asymptotically reach perfection (~100%).

By "monosyllabic" we refer to the words that have only one syllable e.g. Ek, Dui, Tin etc. (in Bangla) or one, good, nice etc. (in English). Monosyllabic means those words that need only one stretch of breath to pronounce. Then, bi-syllabic are words that need two stretches (or puffs) of breaths such as Kori (ko – ri), Kathal (ka – thal), Kormo (kor – mo) etc. in Bangla. Finally, the polysyllabic words that we refer to are words that have more than two syllables. Example: prottutponnomotitto (prot-tut-pon-no-mo-tit-to), Kingkortobbobimurho (king-kor-tob-bo-bi-mur-ho) etc.

We shall discuss the performance analyses (comparatively) of our FIS, ANN based systems and an HMM based speech recognizer and then will follow it up with comparative analysis of the training times of our systems. Finally, we shall comparatively analyze the times for training the systems in regard to altering the data or word descriptions.

4.1. Performance Analysis of our Fuzzy Logic Based System, ANN Based System and an HMM Based System

The monosyllabic, bi-syllabic and polysyllabic distinctions were kept as constant in 5 different test case scenarios. In the following subsections, we analyze the accuracies of our systems as follows: subsection 1 presents the result when the systems are trained by a male voice and tested with a male voice and subsection 2 presents the results when the systems are trained with female voice and tested against a female voice. Subsection 3, however, presents the anomalous case where a female voice is tested when the systems had been trained by a male voice. A similar scenario is presented in subsection 4 where a male speaker was tested on female-trained systems. We had put our systems against an HMM based (phonetic level) speech recognition software – Dragon Naturally Speaking developed by Nuance Communications. Since the commercially developed software is phonetic based, it was language independent, giving us the liberty to test our systems against it but that software gave us transliteration of Bengali words in English rather than UNICODE Bangla text. It should be mentioned that since the commercial HMM based speech recognition software provided no provision for training, the accuracies found through that system were kept constant in all the scenarios. Finally, in subsection 5 we present the results of accuracy when we compared our systems against the commercial HMM based speech recognition software over the complete dataset giving us an aggregation of the results found in the earlier subsections. It is important to mention that all the tests were ten-fold cross validated so that no training data and testing data overlapped and we did not get false or forced positives (elaborated in subsection 3.11.2a). In all the scenarios the 50 Bangla words have been evaluated as the data-set and the results are presented in regard to monosyllabic, bi-syllabic and polysyllabic cases. As justified in section 3.11.2, we have used the training algorithm "traingdx" to train our networks for the comparative recognition accuracy analysis section. It is also important to note that all the tests were coherent using both our Fuzzy logic based system and feed-forward backpropagation ANN based system so that the comparative analysis gives us meaningful information, which evidently is one of the major goals of this research.

1. **Male Voice Trained – Male Speaker Scenario:** The first test case scenario is the first and most general analysis. Here, the training was done by a male speaker and the recognizing systems were tested by a different male speaker. It can be intuitively derived that in this particular case the system gave one of the most optimal accuracies (Table 5).
   From the results (Figure 19) we can see that for optimally trained systems, our ANN based system gives the best performance among the three but if we look into the "polysyllabic" bands/columns in the plot, we notice that the fuzzy logic based system

Table 5. Comparative recognition performance analysis of our fuzzy logic based system, ANN based system and an HMM based speech recognition system for male voice trained – male speaker scenario

System	Accuracy		
	Monosyllabic	Bisyllabic	Polysyllabic
Fuzzy logic based (our system)	65%	75%	90%
ANN based (our system)	94%	89%	78%
HMM based system	78%	73%	70%

surpasses both the ANN and HMM based systems.

2. **Female Voice Trained – Female Speaker Scenario:** This second scenario is similar to the scenario presented in subsection 1. However, it has to be noted that female voice reaches higher frequencies for a particular word than that of a male speaker. The natural average frequency of a male voice is 120 Hz whereas for female voice it is 210 Hz (Traunmüller & Eriksson, 1995). It will become more vivid in subsection 3. Since in this scenario the training and speaking, both have been done by a female speaker (two different speakers but both female), the accuracy reaches a relatively optimal level. The recognition accuracy table and plot are given in Table 6 and Figure 20.
   The findings of this test also coincide with that of subsection 1.
3. **Male Voice Trained – Female Speaker Scenario:** In this scenario, the importance of training the system becomes precise and illustrious. Intuitively, we may concur that if a speech recognition system has been trained and optimized for male voice, it will not perform as well as it would for whom it was trained since the natural frequency range of females are higher (~210Hz) than that of males (~210 Hz) (Traunmüller & Eriksson, 1995). Our result coincides with this fact. The facts and figures are given in Figure 21.

Figure 19. Comparative recognition performance analysis of our fuzzy logic based system, ANN based system and an HMM based speech recognition system

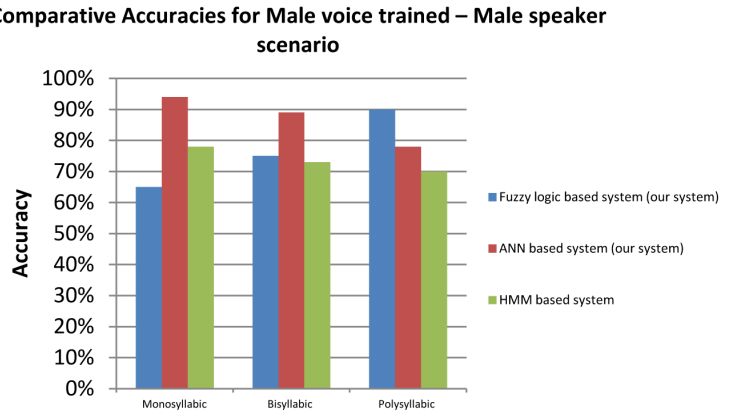


Table 6. Comparative recognition performance analysis of our fuzzy logic based system, ANN based system and an HMM based speech recognition system for Female voice trained – female speaker scenario

System	Accuracy		
	Monosyllabic	Bisyllabic	Polysyllabic
Fuzzy logic based (our system)	70%	75%	85%
ANN based (our system)	92%	86%	72%
HMM based system	78%	73%	70%

Figure 20. Comparative recognition performance analysis of our fuzzy logic based system, ANN based system and an HMM based speech recognition system for female voice trained – female speaker scenario

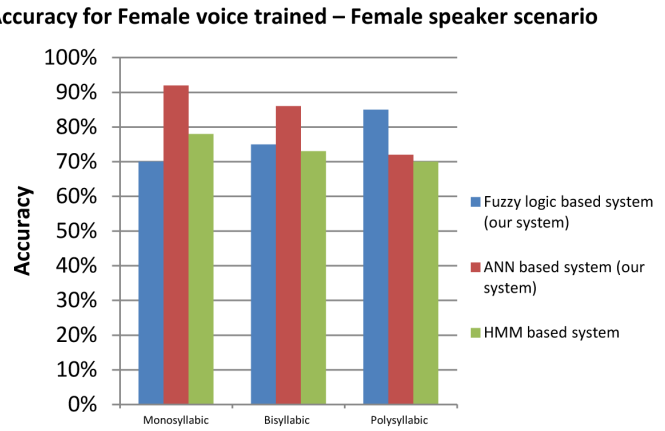
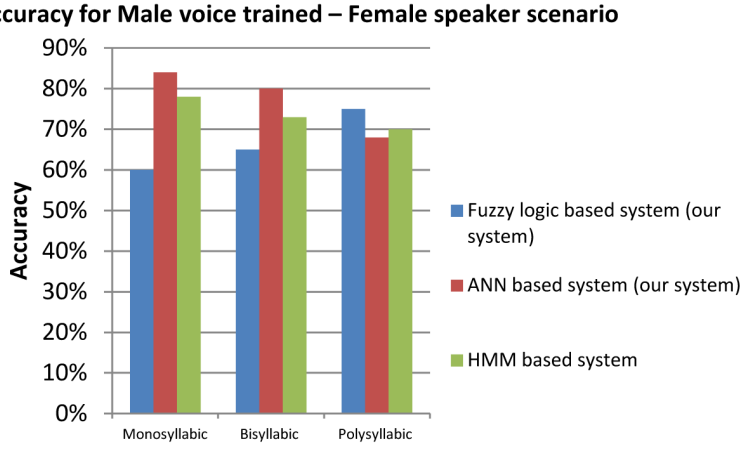


Figure 21. Comparative recognition performance analysis of our fuzzy logic based system, ANN based system and an HMM based speech recognition system for male voice trained – female speaker scenario



The findings (Table 7) of this and the next subsections are important to realize the importance of "training" and "user-adaptiveness" for speech recognition systems (elaborated in section 3.11.1).

4. **Female Voice Trained – Male Speaker Scenario:** This scenario corresponds to the same test case as the one described in subsection 3. Due to the mismatch of frequencies, the system becomes less "speaker adaptive" and the accuracy deters considerably. The findings are presented in Table 8 and Figure 22.

From the Figure 22 we can see that the findings are similar to that of the findings analyzed in subsection 3. The aggregation

Table 7. Comparative recognition performance analysis of our fuzzy logic based system, ANN based system and an HMM based speech recognition system for male voice trained – female speaker scenario

System	Accuracy		
	Monosyllabic	Bisyllabic	Polysyllabic
Fuzzy logic based (our system)	60%	65%	75%
ANN based (our system)	84%	80%	68%
HMM based system	78%	73%	70%

Table 8. Comparative Recognition Performance Analysis of our Fuzzy Logic based system, ANN based system and an HMM based speech recognition system for Female voice trained – Male speaker scenario

System	Accuracy		
	Monosyllabic	Bisyllabic	Polysyllabic
Fuzzy logic based (our system)	65%	65%	75%
ANN based (our system)	88%	82%	72%
HMM based system	78%	73%	70%

Figure 22. Comparative recognition performance analysis of our fuzzy logic based system, ANN based system and an HMM based speech recognition system for Female voice trained – Male speaker scenario

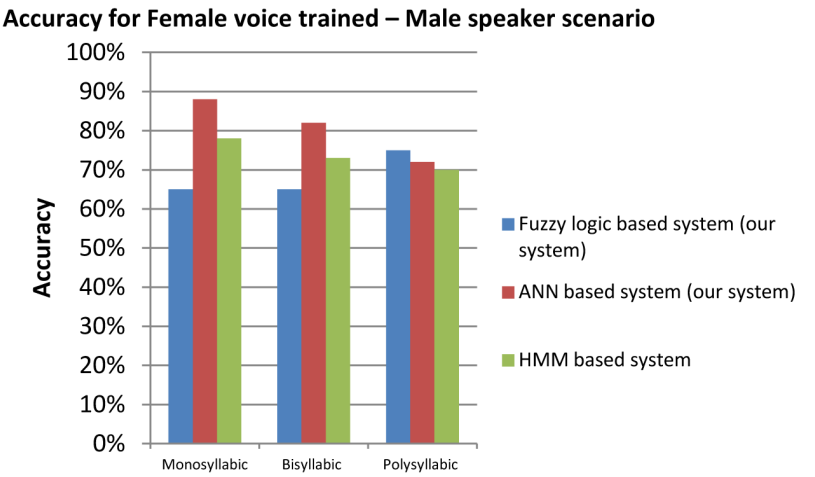


Figure 23. Comparative recognition performance analysis of our fuzzy logic based system, ANN based system and an HMM based speech recognition system over the whole dataset of 1000 word descriptions consisting of 50 words (tenfold cross validated)

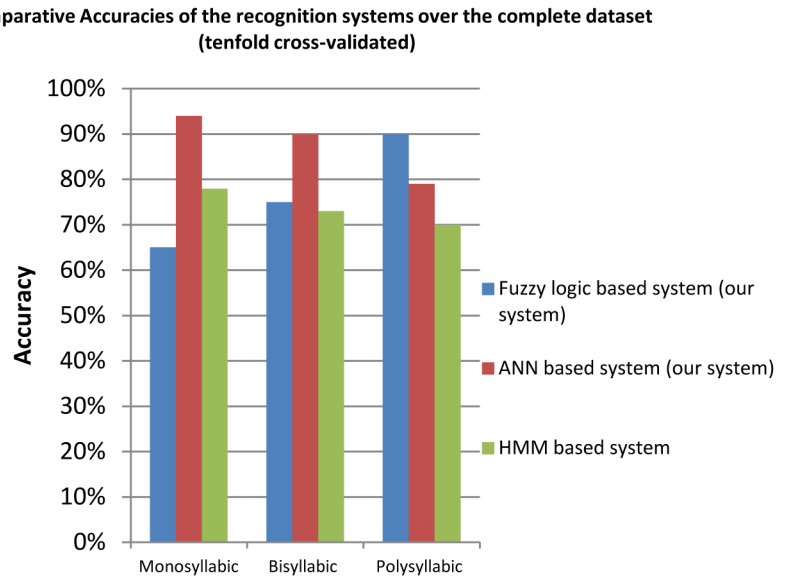


Table 9. Comparative Recognition Performance Analysis of our Fuzzy Logic based system, ANN based system and an HMM based speech recognition system over the whole dataset of 1000 word descriptions consisting of 50 words (tenfold cross validated)

System	Accuracy		
	Monosyllabic	Bisyllabic	Polysyllabic
Fuzzy logic based (our system)	78%	87%	90%
ANN based (our system)	94%	93%	79%
HMM based system	78%	73%	70%

of the findings of subsection 1, 2, 3 and 4 are illustrated in Figure 23.

From the findings of subsection 1, 2, 3 and 4, it is clear that, the more appropriate training the system gets from the speaker, the more user-adaptive it becomes and the accuracy get higher through training. The accuracy rates presented in the paper are the accuracy rates at the time of writing the paper, however, with more training the accuracy rates, can, theoretically, get higher (getting closer to 100% by every training iteration), drastically for a particular speaker.

5. **Comparison of our Systems (FIS and ANN) and an HMM Based Speech Recognition System Over our Complete Dataset (Tenfold Cross Validated):** For comparison purposes, our systems were compared against an HMM based (phonetic level) speech recognition software – Dragon Naturally Speaking developed by Nuance Communications. As the latter was phonetic based and was language in-

dependent, it facilitated the testing but that software gave us transliteration of Bengali words in English rather than UNICODE Bangla text. The accuracy rates that we found are illustrated below in Table 9 and Figure 23. (It is to be noted that the whole dataset of 1000 word descriptions were tested in this scenario through ten-fold cross validation to maintain authenticity of the findings).

The findings of this subsection shed comprehensive light over the research itself and its significance. We can see that our ANN based system performs the best on the average since it gives us recognition performances as 94%, 90% and 79% for Monosyllabic, Bi-Syllabic and Polysyllabic Bangla words respectively. These accuracies are far beyond the HMM based system and somewhat better than the fuzzy logic based system. It is illustrated Figure 24.

However, the most interesting finding from our systems is that the fuzzy logic based recognition recognizes the relatively more difficult Bangla words (polysyllabic) better than both our ANN based system and the commercial HMM based system to a greater extent than that of easier or shorter words (Monosyllabic and Bi-syllabic).

It also coincides to our understanding that our systems give better performance than the HMM based system in Bangla speech since it has been specifically trained for Bangla word recognition and it works on the "word-level" rather than the "phonetic level". Thus, even though the ANN based system wins contest in regard to recognition accuracy based on Figure 24, in reality, for Bangla speech, the fuzzy logic based system performs the recognition more successfully as it handles polysyllabic words better than the other systems. Besides this, the training and altering times come into

Figure 24. Plot of comparative average accuracies based on the whole dataset (tenfold cross validated) derived via averaging the results of Table 9 or Figure 23

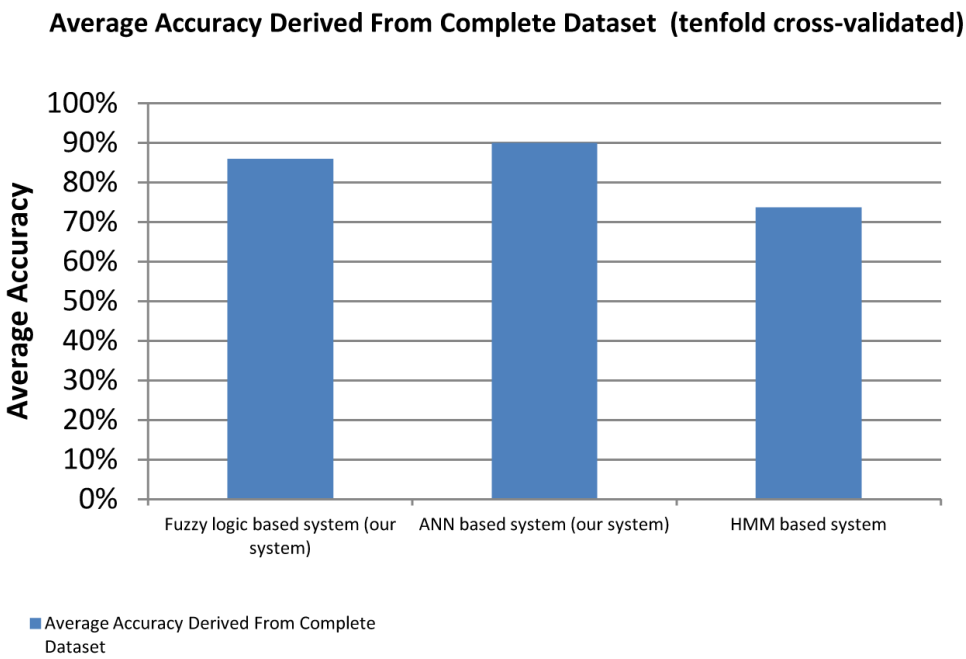
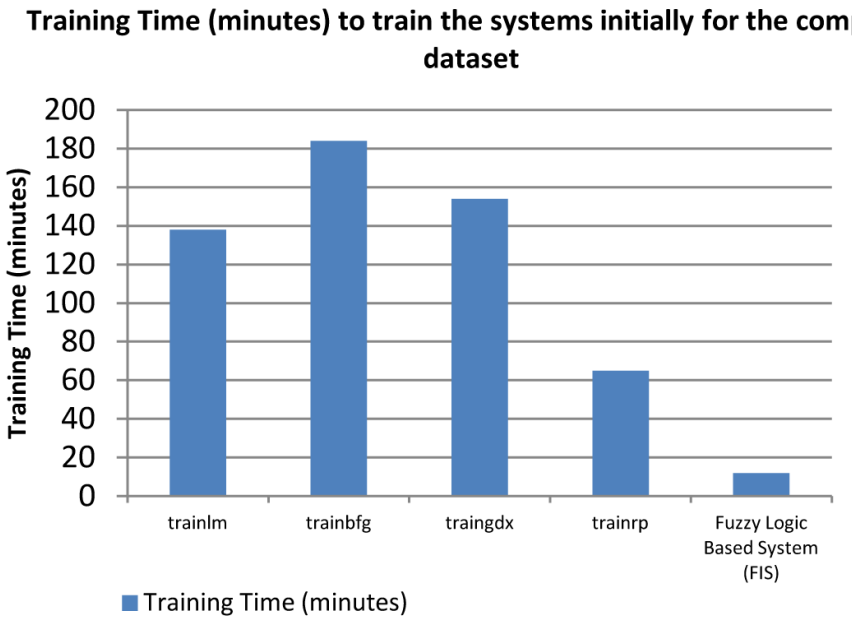


Figure 25. Comparative Initial Training Time for complete dataset



consideration as well and they are discussed in the following subsections.

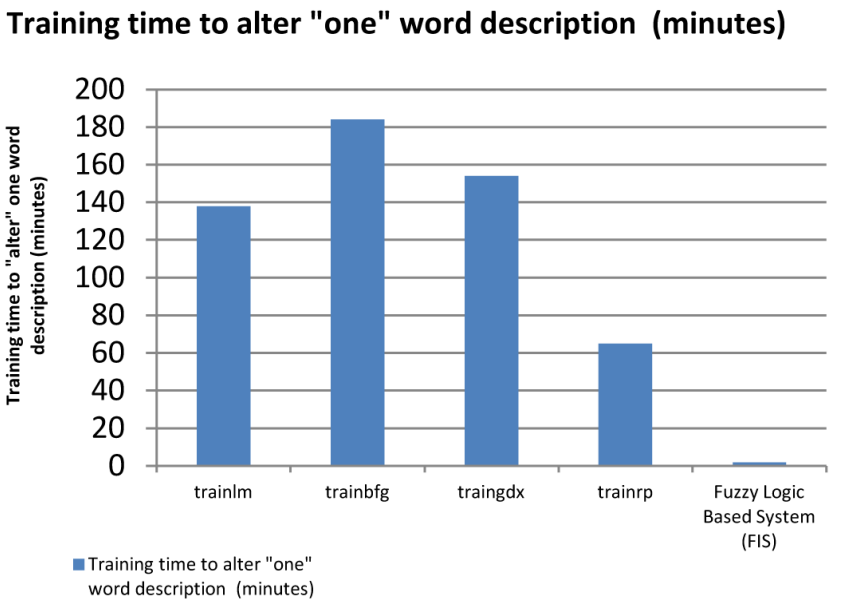
4.2. Comparative (Initial) Training Time Analysis of the Fuzzy Logic Based System and ANN Based System

We had 1000 word descriptions (each having 3 x 25 x 100 = 750 features in each word description from ceptral analysis used in the ANN based system and 30 x 40 = 1200 features from the spectral analysis used in the fuzzy logic based system in each word description). ANN based systems take a considerable amount of time to train and therefore, the training time presents itself as a considerable factor in the development of such expert systems. On the other hand, Fuzzy Logic based systems need only to store the word descriptions in a way such that it can be manipulated in the manner of utility.

We have discussed in section 3.11.2 that we had trained our ANN based system with 4 training algorithms namely trainlm (Levenberg-Marquardt backpropagation), trainbfg (BFGS quasi-Newton backpropagation), traingdx (Gradient descent with momentum and adaptive learning rate backpropagation) and trainrp (Resilient backpropagation) (Mathworks, 2011). From the findings of the training time and the responses of accuracy of the training algorithms we have selected "traingdx" as the optimal training algorithm for our system in regard to comparison to other systems. The following figure illustrates the training time to train the complete dataset including the training time for our fuzzy logic based system.

From the illustration of Figure 25 we can vividly see that the fuzzy logic based system took considerably less time to train itself regardless of the extra number of features for each word description.

Figure 26. Comparative training time plot to alter "one" word description once systems are trained with the complete dataset



4.3. Comparative (Alteration) Training Time Analysis of the Fuzzy Logic Based System and ANN Based System

The previous subsection discussed the training time of the "initial" dataset and we have seen that the durations are quite high (~ 2 hours for each ANN training algorithms). But in this subsection, we shall discuss the training times our systems need to "alter" rather than to store or create the word descriptions. As discussed in subsection 3.11, if the system recognizes a word incorrectly then the user gets a prompt to input the correct word he/she has uttered. Here, we shall present the time the systems need to alter the word descriptions to accommodate such user adaptive behavior.

First, we shall illustrate the training time to alter a (one) word description once the systems have been trained already. The comparative training time plot to alter a word description is given in Figure 26.

As we can see, the training times for the ANN training algorithms remain the same as Figure 25 presented in subsection 4.2. The reason for this is, if we want to alter a single word description in an ANN based system then the whole network needs to be re-trained to calculate the weights and biases for the feed-forward backpropagation architecture which evidently is not time efficient in comparison to the fuzzy system (shown in the plot in Figure 26). However, altering a single word in the FIS refers to simply altering the word description stored in the dataset using Equation (13). Thus, the alteration or updating of the system to make it more adaptive to a particular user happens almost in "real time" as it does not need to modify the rest of the words or classes. Therefore, in regard to "user adaptiveness" and "recognition performance", in the trade-off between performance and accuracy, the fuzzy logic based system evidently performs better even though the ANN based recognition system exhibits higher average recognition accuracy.

5. CONCLUSION

The fuzzy logic based system developed by us is tenably the first speech recognition attempt in Bangla speech using fuzzy logic, even though there have been several attempts using ANN classifiers. Yet again, no attempts using cepstral analysis as features for the ANN was introduced in recognition of Bangla speech till date. However, our systems are not without their limitations. These particular systems could be extended to recognize continuous speech. Moreover, the overall accuracy of the system could be further improved using newer ANN training algorithms. However, we propose that fuzzy logic has to be the base for all linguistic ambiguity-related problems in Bangla as we have empirically showed that the more ambiguous (i.e. Polysyllabic Bangla words in this context) the linguistic entities get in Bangla speech (and other languages of the same family), the better the response we get from fuzzy logic. As an end-note it can be said that speech recognition was an "open" problem before our systems were developed and it remains the same upon completion of the systems – but it is a considerable step in reaching the solution to an "open" problem using spectral analysis, fuzzy logic, cepstral analysis and ANN in Bangla speech recognition.

REFERENCES

Chen, T. (2012). A PCA-FBPN approach for job cycle time estimation in a wafer fabrication factory. *International Journal of Fuzzy System Applications (IJFSA), 2*(2) (2012): 50-67. Web. Retrieved November 7, 2012. doi:10.4018/ijfsa.2012040103

Davies, K. H., Biddulph, R., & Balashek, S. (1952). Automatic speech recognition of spoken digits. *The Journal of the Acoustical Society of America*, *24*(6), 637–642. doi:10.1121/1.1906946

Fletcher, H. (1922). The nature of speech and its interpretation. *Journal of the Franklin Institute*, *193*(6), 729–747. doi:10.1016/S0016-0032(22)90319-9

Hasan, M. R., Nath, B., & Alauddin, B. M. (2003). Bengali phoneme recognition: A new approach. In *Proceedings of the 6th International Conference on Computing and Information Technology (ICCIT) Conference*, Dhaka.

Hasnat, M. A., Jabir, M., & Mumit, K. (2007). Isolated and continuous Bangla speech recognition: Implementation, performance and application perspective. In *Proceedings of the International Symposium on Natural Language Processing (SNLP) 07*, Kasetsart University, Bangok, Thailand.

Illinois Image Formation and Processing (IIFP). (2010). *DSP mini-project: An automatic speaker recognition system.* Retrieved November 7, 2012, from http://www.ifp.illinois.edu/~minhdo/teaching/speaker_recognition/speaker_recognition.html

Islam, M. R., Sohail, A. S. M., Sadid, M. W. H., & Mottalib, A. (2005). Bangla speech recognition using three layer back-propagation neural network. In *Proceedings of the National Conference on Computer Processing of Bangla* (*NCCPB)*, Dhaka.

Jeon, M. J., Lee, S. W., & Bien, Z. (2011). Hand gesture recognition using multivariate fuzzy decision tree and user adaptation. *International Journal of Fuzzy System Applications (IJFSA), 1*(3) (2011): 15-31. Web. Retrieved November 8, 2012. doi:10.4018/ijfsa.2011070102

Juang, B. H., & Rabiner, L. R. (2005). *Automatic speech recognition -a brief history of the technology* (2nd ed.). Amsterdam, Holland: Elsevier Encyclopedia of Language and Linguistics.

Karim, A. H. M. R., Rahman, M. S., & Iqbal, M. Z. (2002). Recognition of spoken letters in Bangla. In *Proceedings of the 5th International Conference on Computing and Information Technology (ICCIT) conference*, Dhaka.

Mathworks. (2011). *MATLAB Neural network toolbox documentation.* Retrieved November 2, 2012, from http://www.mathworks.com/help/pdf_doc/nnet/nnet_ug.pdf

Molau, S., Pitz, M., Schlüter, R., & Ney, H. (2001). Computing mel-frequency cepstral coefficients on the power spectrum. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Salt Lake City, UT (pp. 73-76).

Plannerer, B. (2005). *An introduction to speech recognition.* Retrieved November 2 2012, from http://www.speech-recognition.de/pdf/introSR.pdf

Rahman, K. J., Hossain, M. A., Das, D., Islam, T. A. Z., & Ali, M. G. (2003). Continuous Bangla speech recognition system. In *Proceedings of the 6th Int. Conf. on Computer and Information Technology (ICCIT) Conference,* Dhaka.

Roy, K., Das, D., & Ali, M. G. (2002). Development of the speech recognition system using artificial neural network. In *Proceedings of the 5th International Conference on Computing and Information Technology (ICCIT) Conference*, Dhaka.

Smith, J. O. (2007). Spectrograms. Mathematics of the Discrete Fourier Transform (DFT), with Audio Applications. W3K Publishing. ISBN 978-0-9745607-4-8.

Traunmüller, H., & Eriksson, A. (1995). *Publications of Hartmut Traunmüller*. Stockholm University, Sweden. Retrieved October 30, 2012 from http://www.ling.su.se/staff/hartmut/f0_m&f.pdf

Traux, B. (Ed.). (1999). *Mel. Handbook for acoustic ecology*. Simon Fraser University and ARC Publications.

Weiss, M. (2006). Indo-European language and culture. *Journal of the American Oriental Society*. Retrieved September 24, 2012, from http://findarticles.com/p/articles/mi_go2081/is_2_126/ai_n29428508/

Yang, S., Park, K., & Bien, Z. (2012). Gesture spotting using fuzzy garbage model and user adaptation. [IJFSA]. *International Journal of Fuzzy System Applications*, *1*(3), 47–65. Retrieved November 7, 2012. doi:10.4018/ijfsa.2011070104

Adnan Firoze has completed his Bachelor of Science (B.S.) degree in Computer Science and Engineering in 2011 from North South University, Bangladesh. He received the distinction of being the highest CGPA achiever among all CSE majors and was awarded the Summa Cum Laude award. His research interests include Audio and Speech processing, Fuzzy systems, Computational Linguistics, Image Processing, Game Theory and Econometrics. He believes in an interdisciplinary approach to Computer Science where the core discipline meets diverse fields such as Economics, Social Networking, Music, Medicine and Linguistics. Apart from his successful major in Computer Science, he holds a minor in English with concentration on Linguistics. Additionally, he is interested in Journalism and pursues it actively. He is affiliated with IEEE and United Nations Youth & Students Association of Bangladesh. His resolution lies in research in the intersection between Computer Science and the humanities.

Md Shamsul Arifin has graduated from North South University with a BS in Computer Engineering. He has worked on Rich Communication System and NAT Traversal in Mobile Platform, Speech Processing, Image Processing, Fuzzy Logic and Neural Networks. His research interests include Intelligent Systems, Bioinformatics and Human Computer Interaction. He is a former member of North South University Computer Club and has organized several workshops, seminars and programming contests including ACM-ICPC regional. At present he is working as Research Assistant in North South University.

Rashedur M. Rahman is working as an Associate Professor in Electrical Engineering and Computer Science Department in North South University, Dhaka, Bangladesh. He received his Ph.D. in Computer Science from University of Calgary, Canada and Masters from University of Manitoba, Canada in 2007 and 2003 respectively. He has authored more than 50 peer reviewed journals and conference proceedings in the area of parallel, distributed, grid and cloud computing, knowledge and data engineering. His current research interest is in data mining particularly on financial, medical and educational data, data replication on grid, cloud load characterization, optimization of cloud resource placements and computational finance. He has been serving in the editorial board of a number of journals in the knowledge and data engineering filed. He also serves as a member of organizing committee of different international conferences.