Feature Extraction and Recognition of Hindi Spoken Words using Neural Networks

Poonam Sharma
Assistant Professor
Department of Computer Science
The NORTHCAP University,
Gurgaon 122001, India

Anjali Garg
Department of Computer Science
The NORTHCAP University,
Gurgaon 122001, India

ABSTRACT
Automatic Speech Recognition System has been a challenging and interesting area of research in last decades. Only a few researchers have worked on Hindi and other Indian languages. In this paper, a Speech Recognition System for Hindi language based on MFCC, PLP and neural networks is proposed and it was observed that the accuracy of the system was better than other conventional methods.

Keywords
Automatic Speech Recognition, Mel frequency Cepstral Coefficient, Predictive Linear Coding

1. INTRODUCTION
Speech recognition is a popular research area since a long time. In order to communicate with machine, speech is probably a most efficient and useful interface to interact with. However research work on this technology has been started since 1920. Even after decades of research in this area and many successful commercial products, the performance of speech recognition system lags behind human level performance. A lot of research experiments and results are achieved in English language throughout the world but a limited success is achieved for Hindi Speech recognition. Hindi is the fourth-most spoken native language in the world. Speech is considered as a short-time stationary signal that remains stationary only for short duration. In our system, we have recorded speech from five different speakers, which may be an early attempt for speaker independent isolated Hindi speech recognition. The features are extracted from input signal and are compared with voice template that is stored in computer database in speech recognition system. Later on, classification is done based on acoustic properties of word. In order to increase recognition rate and improve recognition results, neural networks are used.

In recent years a huge effort is made by many researchers in the field of speech recognition but it would be extremely significant if speech based interface could interact with machine in native idiom. In India there are 22 executive languages and 419 other livelihood languages. Majority of population is unaware of English language either in reading or writing but they can take advantage from resources of Information Technology sector if they support native idiom. Therefore, there is huge scope to develop such systems in Hindi language.

The paper is organized as: Section 2 describes the data collection module whereas Section 3 describes how the features are extracted. Section 4 focuses on various recognition techniques and later on in Section 5, proposed algorithm is described followed by flowchart in Section 6. Section 7 describes the output of algorithm with simulated results in Section 8. Finally conclusion is presented in section 9.

2. DATA COLLECTION MODULE
Sound in general, is a vibration that is transmitted as a mechanical wave with the rate of pressure variation from low and high, therefore determining frequency. The difference between the low and the high frequency determines amplitude. Normally, the speech is stored as vector of samples, where each value is a double-precision floating point number. The sequence of these numbers in addition to sampling rate defines the complete sampled sound.

The Coles is a tool that is used for recording sound for this particular work. While using this tool user has to specify the sampling frequency that tells how many samples an audio carries per second, number of bits that are used to store sample value that can be 8 or 16 that depends on storage requirement of samples as a floating number and filename.

Here in this particular work, the sampling frequency of 16 KHz is used for recording sound with number of bits as 16 and recordings are saved with .wave file extension. Here 100 words are spoken in Hindi language by two speakers (1 male and 1 female). The below is the two-dimensional graph of plotting a speech with number of samples at abscissa and amplitude at ordinate.

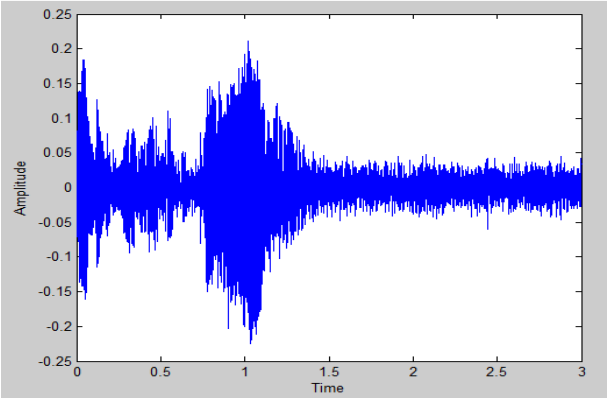

Fig 1: Speech signal of word "kaam"

3. FEATURE EXTRACTION MODULE
This phase allows classification of sounds by deriving descriptive features from pre-emphasized speech signal. Feature extraction step generate parameters that are derived from mathematical representation of speech signal waveform. There parameters are known as feature vectors. Basically, feature extraction techniques are categorized as (1) temporal analysis where speech waveform itself is analyzed and (2) spectral analysis techniques where spectral representation of speech is analyzed. The various techniques for feature extraction are follows:

3.1 Mel Frequency Cepstral Coefficient (MFCC)

Davis and Mermelstein gave this beneficial approach for speech recognition.

The foremost point to be aware of speech is that the sounds produced by human are passed through a filter of shape of vocal tract that includes tongue, teeth etc. The sound that comes out is determined by this shape. If the shape is accurately determined, the accurate representation of phoneme that is being produced is obtained. The shape of vocal tract itself is the envelope of short time power spectrum. To represent this envelope accurately is the task of MFCC.

Pre-emphasized acoustic wave is given as an input for feature extraction processing [1]. Under this, various steps are followed whose detailed description is given below:

3.1.1 Frame the Pre-Emphasized Speech Signal into Short Frames

A speech is non-stationary signal but it is assumed as stationary on a short time scales in order to simplify things though samples are changing constantly even on short time scales. For this, signal is divided into 20-40ms frames. If the frame length is too short, then resolution of narrow band components is sacrificed that affects frequency resolution and if it is longer, signal properties changes too much through the whole frame that affects time resolution [2]. Therefore 25ms is standard.

3.1.2 DFT Computation

At first signal is in time-domain. By discrete Fourier transformation, we convert it into frequency domain. This computation is done for the next step to obtain spectral information. The computation of Discrete Fourier Transform [3] is given by:

$$S_i(k) = \sum_{n=1}^{N} S_i(n)\, h(n)\, e^{-j2\pi k n / N} \qquad (1)$$

Where $S_i(k)$ represents equivalent DFT

and *i* represents frame number

The output of DFT is a complex number. The complex data is ignored [4] as speech recognition system deals only with real data.

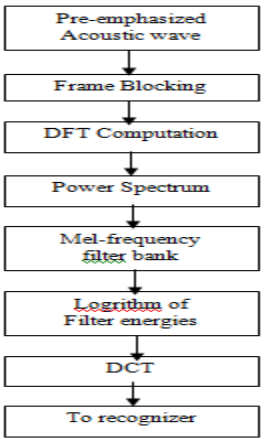

Fig 2 : Block diagram of MFCC feature extraction processing

3.1.3 Compute Periodogram Estimate of Power Spectrum for each Frame

This step is inspired by cochlea, an organ in human ear. Depending on the frequency of input sound, cochlea vibrates at different location. Therefore, depending upon this location nerves sends information to the brain about the presence of frequency. The periodogram estimate does this task and identifies different frequencies present in the frame. To compute periodogram estimate of power spectrum for speech frame s_i (n), perform the following:

$$P_i(k) = |S_i(k)|^2 / N \qquad (2)$$

Where, Pi (k) represents power spectrum of frame i.

This is called Periodogram- based power spectral estimation where absolute value of DFT is taken and the result is squared.

3.1.4 Compute the Mel Frequency Filter Bank

The periodogram spectral estimate holds much information that is unnecessary for speech recognition system. The cochlea cannot differentiate between two narrowly spaced frequencies. As the frequency increases this effect becomes more pronounced. Therefore, cluster of periodogram bins is taken and they are summed up to discover amount of energy persist in each frequency region. Mel-filter bank performs this job. More often triangular filter bank is used for this operation; where first filter is extremely narrow and indicates the amount of energy persist near 0 Hertz. In practical, negligible energy exists at 0 Hertz. Hence first filter is ignored for this particular task. As the triangular word suggest, with increase in frequency, the filters get wider. In general, in low frequency region more number of filters is required whereas in high frequency region [5], less number of filters are required. Basically, Mel Filter bank indicates how much energy exists at each frequency. The Mel scale is helpful to find how filter banks are spaced and their width.

In general a set of 20-40 triangular filters are used and are applied on the periodogram-based power spectral estimate that is obtained from step 3. The filter bank energies (FBEs) are calculated by multiplying each filter bank with power spectrum and then coefficients are added. The below graph is a plot of Mel Filter bank and power spectrum.

3.1.5 Take the logarithm of FBEs

Once the filter bank energies are obtained, their logarithm is taken. This is inspired by human perception. As human ear works in decibel [6], so it is necessary to take log of FBEs. Also the speech signal does not go after linear frequency as used in FFT. Therefore Mel scale is used for this processing. The frequency of Mel scale is proportional to logarithm of frequency of linear scale that reflects human hearing Also the triangular filter bank used in step 4 are non-uniformly spaced on linear scale and uniformly spaced on Mel scale. The formula for converting linear frequency to mel frequency is:

$$M(f) = 1125 \ln (1 + f / 700) \qquad (3)$$

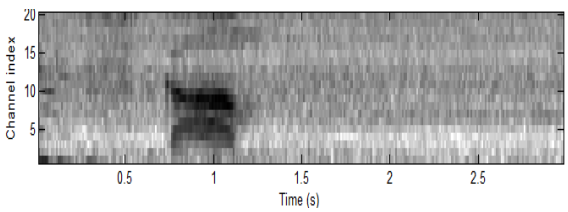

Fig 3: Log (mel) Filterbank Energies

3.1.6 Take DCT of logFBEs

This is the final step of feature extraction processing where the Discrete Cosine Transform of log filter bank energies is computed. FBEs are relatively correlated with each other because all filter banks are overlapping. Therefore, to decorrelate the energy DCT is used. For speech recognition system, the lower 12-13 coefficients are kept, rest are discarded because higher value of DCT coefficient shows rapid changes in filter bank energies which in turn degrades the performance of ASR system.

The resulting features are called Mel Frequency Cepstral Coefficient (MFCC). These features are sent to the recognizer for actual computation of word sequence.

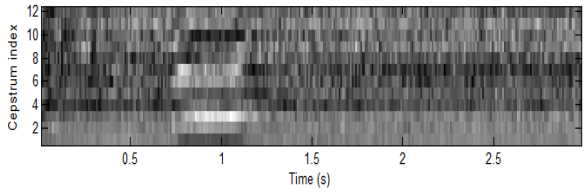

Fig 4: Mel Frequency cepstrum

3.2 Perceptual Linear Prediction (PLP):

The PLP model is proposed by Hynek Hermansky in 1990. It describes psychophysics of human hearing in a better way. It reduces word error rate by removing irrelevant information of speech. PLP is similar to LPC analysis except spectral characteristics of signal are transformed so as to match with characteristics of human auditory system [7]. The below are the steps for extracting features using PLP

At first, speech is given as an input to machine and perform RASTA-PLP (Relative Spectra filtering) [8], Calculate PLP cepstra and spectra. For this, follow the below steps:

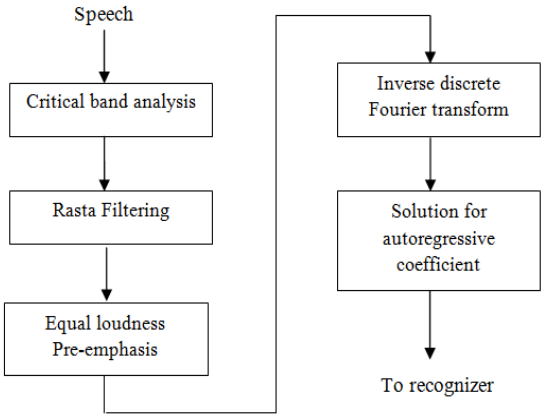

Fig 5: Block diagram of PLP

3.2.1 Compute power spectrum of input

At first power spectal density of given frame is computed. Basically, a power spectrum distributes the energy of input waveform to various frequency components. A measure of power intensity of input signal in frequency domain is called Power Spectral Density (PSD). The Fast Fourier Transform (FFT) spectrum of a signal is used to compute the PSD. It provides a way to describe frequency and amplitude of input signal.

3.2.2 Perform critical band analysis and put it under log domain

A critical band is a frequency bandwidth of auditory filters, which is created by a sense organ of hearing in human ear named cochlea. In other words, critical band is a band of audio frequencies, where with the use of auditory masking, the second tone will interfere with the perceptron of first one. For this, the power spectrum obtained in above step is wrapped onto Bark scale, a frequency scale where equal distances correspond with perceptually equal distances whose range is 1 to 24. The bark scale is combined with power spectral of critical band filter and simulates the frequency resolution of human ear that is nearly constant on the Bark scale. This combination decreases spectral resolution.

3.2.3 Perform RASTA filtering

Relative Spectral filtering uses band pass filtering of energy in each frequency sub band in log spectral domain. This removes slow channel variations and removes constant offset obtained from static spectral coloration in speech channel. It is applied to cepstrum for both spectral and cepstral filtering.

3.2.4 Do final auditory compression

This includes loudness equalization; there is a need of compensating non-equal perception of loudness at various frequencies. Therefore it is pre-emphasized by equal loudness curve, which is a measure of sound pressure over frequency spectrum. Due to this, listener receives constant loudness when spoken in a pure steady tone. Also, auditory compression removes bands those are duplicated, weigh the critical bands, and replicate first and last band.

3.2.5 Perform LPC analysis

Linear Predictive Coding analysis computes autoregressive model from spectral magnitude samples. For this, first apply inverse discrete fourier transform to obtain equivalent autocorrelation function. By this, a value from time series is represented on preceding value from same time series and the output variable linearly depends on preceding value. After that, fit a linear prediction in order to form a model of linear time-invariant digital signal by observing input and output sequence. The most commonly formed model is all-pole model.

3.2.6 Convert LPC to cepstra.

In this step, the lpc ‘a’ coefficient is converted into frames of cepstra.

3.2.7 Convert LPC to spectra.

Here linear predictive coefficients are converted back to spectra and at the output, number of frequency channels are obtained.

3.2.8 Calculate cepstra from spectral samples

There is no need of LPC smoothing of spectrum and cepstra is calculated from spectral samples and discrete cosine transform (DCT) matrix is made.

3.2.9 Apply lifter to the matrix of cepstra

Lifter is applied to the matrix of cepstra which is HTK-style sine-curve liftering.

4. RECOGNITION MODULE

There are several approaches for speech recognition such as Acoustic Modeling approach, Knowledge-based approach, Artificial Intelligence approach. Neural Networks is an attractive acoustic modeling approach since many years in speech recognition system. It is a tool that is used in many

aspects such as isolated word speech recognition, speaker adaptation and to learn relationships between phonemes. Unlike HMM, it does not make any assumption regarding statistical properties of features. Rather it has several qualities that make it a attractive approach for speech recognition. In Matlab, Neural Network Toolbox provides various functions, algorithm, apps in order to create, visualize, train and simulate neural network. The various types of neural network under which these operations are performed are mentioned below:

4.1 Feed Forward Back propagation Network

It is an artificial neural network in which connections between different units do not form cycle. Here, the information moves in single direction i.e. forward from input nodes to output nodes with intermediate hidden nodes in between resulting no cycle in the network. Back Propagation network is a two-layer feed forward network with hidden and output neurons, having enough neurons in hidden layer to classify vectors well. Back propagation itself means "back propagation of errors". It requires a target value for each input value and then output are compared with the targets. If the difference comes, then network is back propagated to the last stable stage. Therefore, it is referred as supervised learning method. The network is trained well by scaled conjugate gradient back propagation.

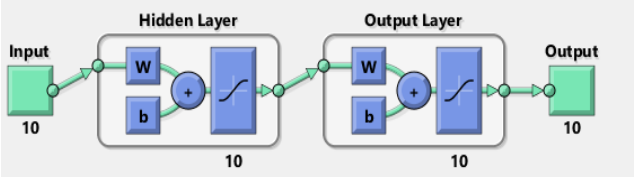

Fig 6: Feed Forward Back Propagation Network

4.2 Perceptron Neural Network:

Rosenblatt gave the several variations of perceptron. The simplest is a single-layer perceptron where the weights and biases are trained to give a correct target vector when corresponding input vector is given. The perceptron learning rule is a training technique used here. It has an ability of generalizing from its training vectors and learning from initial randomly distributed connections. The hard-limit transfer function is used by a perceptron neuron but it allows only 0 or 1 value at the output. Therefore, it is a limitation of perceptron. Also, it classifies only linearly separable set of vectors.

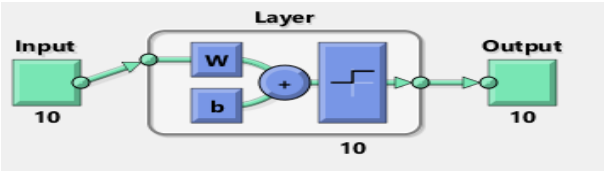

Fig 7: Perceptron Neural Network

4.3 Learning Vector Quantization (LVQ) Neural Network

In this network, first is competitive layer and the second one is linear layer. Competitive layer classify the input vectors into different classes and linear layer transform those classes into target classification as given by user. The classes of competitive layer are known as subclasses and the classes learned by linear are referred to as target classes. Both these layers have one neuron per class. To create a network lvqnet function is used.

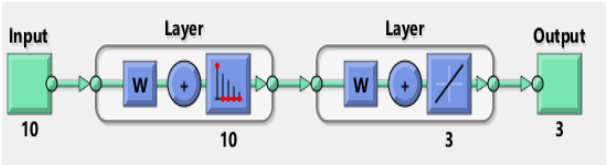

Fig 8: LVQ Neural Network

4.4 Linear Neural Network

The linear network is similar to perceptron but unlike perceptron its transfer function is linear. Therefore, it allows output to take any value not 0 or 1 only. Like the perceptron, it can solve linear separable problems. The supervised learning is used here and difference between input and output vector is called the error. To create a network, newlin function is used that created linear layer for specific purpose.

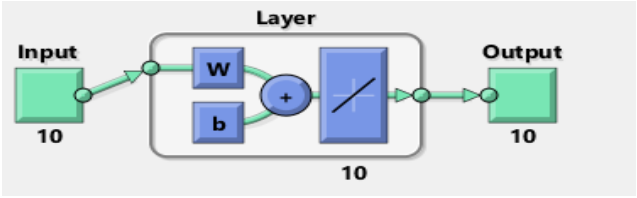

Fig 9: Linear Neural Network

5. PROPOSED ALGORITHM

The algorithm is designed for recognizing Hindi spoken words based on the results obtained from different feature extraction techniques. The proposed algorithm is implemented on Matlab 2012a. The following steps are followed for recognizing speech:

Step 1: Input the speech signal x_i and read sampling rate, number of bits from file.

Step 2: Do windowing by hamming window of length 240 and convert signal from time-domain to frequency domain by applying DFT

$$w(n) = 0.54 - 0.46 \cos(2\pi n / N), \; 0 \le n \le N$$

$$X(k) = \sum_{n=0}^{N} x(n) e^{-j 2\pi k n / N}, \; 0 \le k \le N\text{-}1$$

Step 3: Calculate Mel frequency cepstral coefficient (MFCC) with Mel frequency as given below

Step 4: Calculate Perceptual Linear Prediction (PLP).

Step 5: The final input vector obtained after merging features from step 3 and step 4 is:

$$F = [M_L \; P_L]$$

Step 6: Make target vector.

Step 7: Import final input vector and target vector and create Feed-Forward Back Propagation Network by using nntool.

Step 8: Train the network and simulate results.

Step 9: Test the selected sample against Neural Network.

Step 10: If word spoken correctly, then Speech Recognized and display CORRECT;

else, Speech Not Recognized and display INCORRECT.

Step 11: Plot performance plot.

6. FLOWCHART

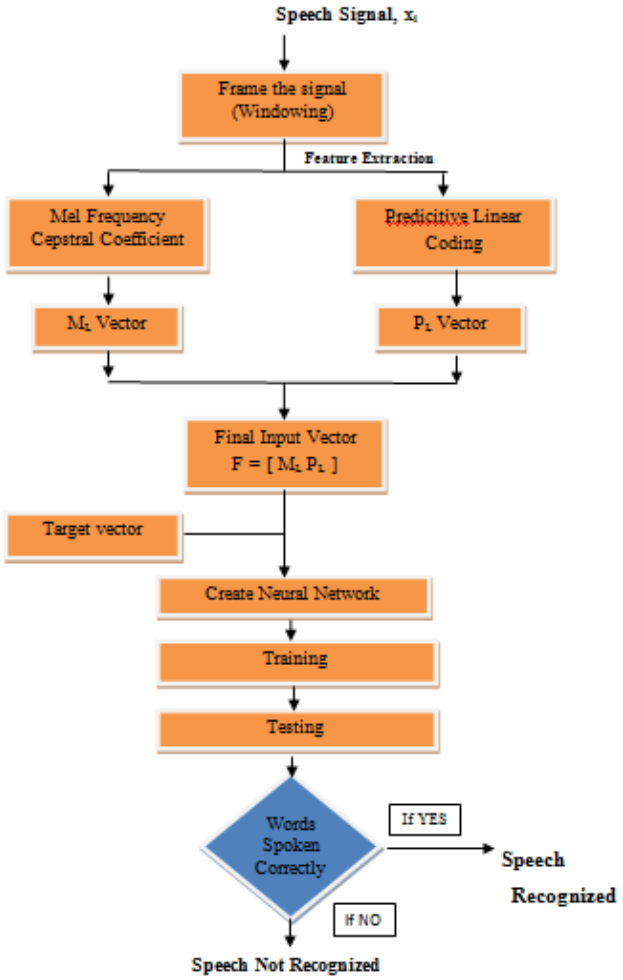

Fig 10: Flowchart

7. OUTPUT OF ALGORITHM

An average accuracy of 78% was found for male speaker whereas 80% for female speaker. Therefore, 79% of average recognition result is obtained as an output of algorithm. Also, the plot of Neural Network Training Performance of female speaker is shown below. The best Validation Performance achieved is 0.045491 at epoch 9.

Table 1: Output of algorithm

	Words Spoken	**Words recognized Correctly**	**Average accuracy**
First speaker (Male)	50	39	78%
Second speaker (Female)	50	40	80%
Output of algorithm			79%

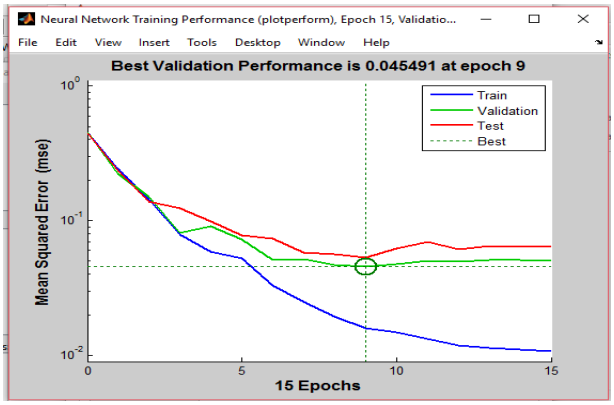

Fig 11: Neural Network Training Performance Plot of "female speaker"

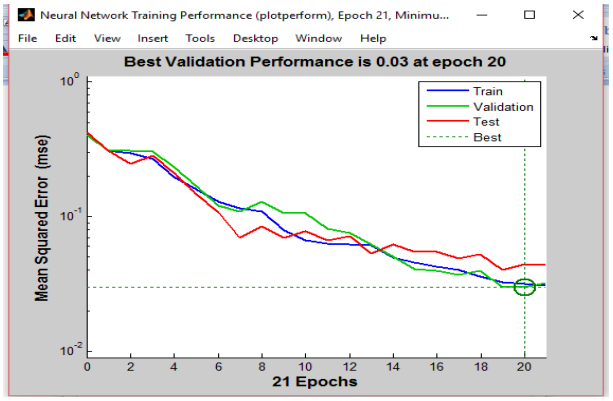

Fig 12: Neural Network Training Performance Plot of "male speaker"

8. SIMULATED RESULTS

Modern speech recognition system uses combination of standard techniques in order to improve performance and accuracy over basic approach. The following table shows that by using the combination of MFCC-PLP the accuracy is improved.

Comparison Of Various Feature Extraction Techniques

Table 2: Comparison of feature extraction technique

SNo.	**Technique**	**Average Accuracy**
1.	Mel-Frequency Cepstral Coefficient (MFCC)	78%
2.	Perceptual Linear Predictive (PLP)	60%
3.	Mel-Frequency Cepstral Coefficient – Perceptual Linear Predictive (MFCC-PLP)	79%

It is clear from the above comparisons that using the combination (MFCC-PLP) we get highest accuracy.

Also, the samples are tested against different neural network and following results are obtained:

Comparison Of Different Neural Networks

Table 3. Comparison of different Neural Networks

SNo.	Technique	Average Accuracy
1.	Feed-Forward BPN	79%
2.	Perceptron Neural Network	73%
3.	Linear Neural Network	69%

It is clear from the above comparisons that the best results are obtained by using Feed Forward Network.

9. CONCLUSION

In this paper a method for Hindi Word Recognition based on MFCC and PLP features was proposed. The proposed method was tested against different neural network techniques and it was observed that the method was giving better accuracies as compared to other conventional methods like HMM and SVM.

10. REFERENCES

[1] Vibha Tiwari. 2010. MFCC and its applications in speaker recognition. International Journal on Emerging Technologies 1(1): (2010), pp. 19-22.

[2] Lindasalwa Muda, Mumtaj Begam and I. Elamvazuthi,. 2010. Voice recognition algorithm using MFCC & DTW techniques. Journal Of Computing, Volume 2, Issue 3, March 2010, ISSN 2151-9617, pp. 138-143.

[3] Ripul Gupta. Speech recognition for Hindi. Indian Institute of Technology, Bombay, pp. 11-14.

[4] Yuan Meng. 2004. Speech recognition on DSP: Algorithm optimization and performance analysis. The Chinese university of Hong Kong, July 2004, pp. 1-18.

[5] Sirko Molau, Michael Pitz, Ralf Schl¨uter, and Hermann Ney. Computing Mel-frequency cepstral coefficients on the power spectrum. University of Technology, 52056 Aachen, Germany

[6] Siddhant C. Joshi, Dr. A.N.Cheeran. 2014. MATLAB Based Feature Extraction Using Mel Frequency Cepstrum Coefficients for Automatic Speech Recognition. International Journal of Science, Engineering and Technology Research (IJSETR), Volume 3, Issue 6, June 2014

[7] Namrata Dave. 2013. Feature Extraction Methods LPC, PLP and MFCC In Speech Recognition. www.ijaret.org Volume 1, Issue VI, July 2013

[8] H. Hermansky and N. Morgan. 1994. RASTA processing of speech. IEEE Trans. on Speech and Audio Proc., vol. 2, no. 4, pp. 578-589, Oct. 1994