

# A comparative study of different features for isolated spoken word recognition using HMM with reference to Assamese language

Sruti Sruba Bharali<sup>1</sup> · Sanjib Kr. Kalita<sup>1</sup>

Received: 6 December 2014 / Accepted: 13 October 2015 / Published online: 19 October 2015  
© Springer Science+Business Media New York 2015

**Abstract** This paper describes the work done in implementation of speaker independent, isolated word recognizer for Assamese language. Linear predictive coding (LPC) analysis, LPC cepstral coefficients (LPCEPSTRA), linear mel-filter bank channel outputs and mel frequency cepstral coefficients (MFCC) are used to get the acoustical features. The hidden Markov model toolkit (HTK) using the Hidden Markov Model (HMM) has been used to build the different recognition models. The speech recognition model is trained for 10 Assamese words representing the digits from 0 (shounya) to 9 (no) in the Assamese language using fifteen speakers. Different models were created for each word which varied on the number of input feature values and the number of hidden states. The system obtained a maximum accuracy of 80 % for 39 MFCC features and a 7 state HMM model with 5 hidden states for a system with clean data and a maximum accuracy of 95 % for 26 LPCEPSTRA features and a 7 state HMM model with 5 hidden states for a system with noisy data.

**Keywords** Speech recognition · Isolated word · Speaker independent · LPC · LPCEPSTRA · MELSPEC · MFCC · HMM · HTK · Assamese language

## 1 Introduction

Speech recognition is one of the many areas in current research explorations. It involves taking speech signal as input and to interpret its meaning. This task can divide into four different categories based on its working principle. The first category is of isolated word recognition which involves recognizing words that are uttered as an individual word. The second category involves continuous word recognition where the time space between words is minimal. The third category is continuous speech recognition where instead of just words whole sentences can also be recognized. The fourth category involves spontaneous speech where along with whole sentences human speech imperfections are also considered. In this paper an attempt is made to build a first category isolated word recognizer.

### 1.1 Motivation

Literature review reveals that studies on the acoustic features of Indian language like Tamil, Malayalam, Telugu, Bengali, Hindi is carried out by several researchers (Bourlard and Morgan 1998; Pruthi et al. 2000; Kumar et al. 2012; Mohamed and Nair 2012; Mehta et al. 2013; Krishna et al. 2014; Mankala et al. 2014). But less effort has been put to develop speech processing tools for North Eastern languages. The lack of effective research work in Assamese has motivated us to develop a speaker independent recognition model. This paper makes an effort to bridge the gap between speech recognition and Assamese language by developing a speaker independent isolated word recognition model with reference to Assamese language. The structural design of such a system is depicted in Fig. 1.

---

✉ Sruti Sruba Bharali  
sruti.s.bharali@gmail.com

Sanjib Kr. Kalita  
sanjib959@gmail.com

<sup>1</sup> Gauhati University, Guwahati, Assam, India

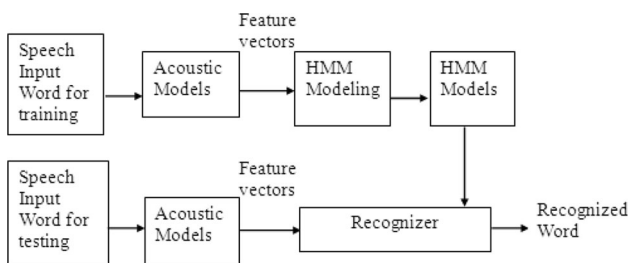


Fig. 1 Block diagram of our Assamese isolated speech recognizer

## 2 Theoretical background

### 2.1 Linear predictive coding (LPC) analysis

Linear predictive coding (LPC) analysis tries to find a speech sample at time  $n$ , when the linear combination of past speech samples are known. LPC can be computed using the following steps: *Pre-emphasis* This is the first step where the speech signal is sent to a low order digital filter. *Frame blocking* The second step involves segmenting the speech signal into successive frames overlapping each other. *Windowing* In this step windowing function is multiplied with each frame. *Autocorrelation analysis* In this step each frame is auto correlated with a value of  $p$ , where  $p$  is the order of LPC analysis. *LPC Analysis* In this step the Durbin’s method is used to convert each frame of  $p + 1$  auto correlations to LPC parameters as in (Eslam Mansour Mohammed et al. 2013). The following Fig. 2 shows LPC feature extraction phase.

### 2.2 LPC cepstral coefficients (LPCEPSTRA)

LPC cepstral coefficients (LPCEPSTRA) is a feature extraction technique that is widely used for speech recognition. LPCEPSTRA can be computed using the similar steps as in LPC analysis where the first five steps are same, followed by a few more steps which are:

*Cepstral analysis* This is the next step after LPC analysis is to derive the cepstrum of LPC using a set of recursive procedures. These cepstral coefficients known as LPC Cepstral Coefficients or LPCEPSTRA and the feature extraction are depicted in Fig. 3.

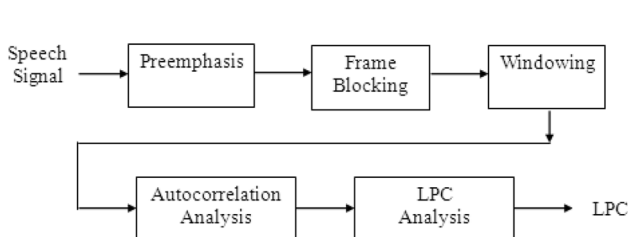


Fig. 2 Linear predictive coefficients (LPC) feature extraction process

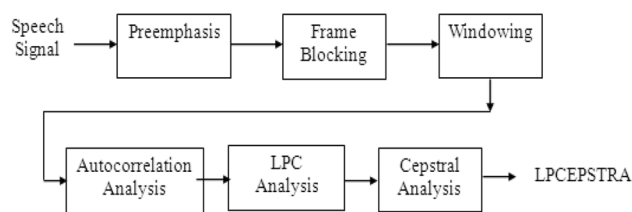


Fig. 3 LPC cepstral coefficients (LPCEPSTRA) feature extraction process

### 2.3 Linear mel-filter bank channel outputs (MELSPEC)

MELSPEC is another technique of feature extraction in speech recognition process. MELSPEC can be computed using the following steps where the first three steps are same as compared to LPC: *Pre-emphasis* This is the first step where the speech sample is sent to a high pass filter. *Frame blocking* This step involves segmenting speech signal into successive frames overlapping each other. *Windowing* In this step windowing function is multiplied with each frame. *MELSPEC analysis* In this step, the Mel frequency filter bank is used to convert the linear frequency to mel frequency using the mathematical relationship formula, where  $f$  is linear frequency.

$$Freq(Mel) = 2595 \times \log_{10}(1 + f/700)$$

A Mel scale frequency is computed, using

$$\Delta f(Mel) = (f_{Hmel} - f_{Lmel}) / (M + 1)$$

The MELSPEC feature extraction process can be shown as in Fig. 4.

### 2.4 Mel frequency cepstral coefficient (MFCC)

MFCC is another technique used in speech recognition for feature extraction process. MFCC can be computed using the following steps where the first four steps are same as compared to MELSPEC followed by two more steps. The additional steps to derive MFCC from MELSPEC are: *MFCC Analysis* In this step, a vector of acoustical coefficients known as MFCC is extracted from each windowed frame after performing fast fourier transform (FFT) and discrete cosine

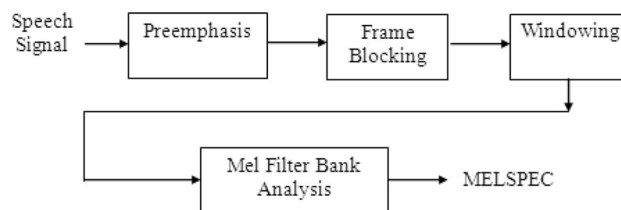
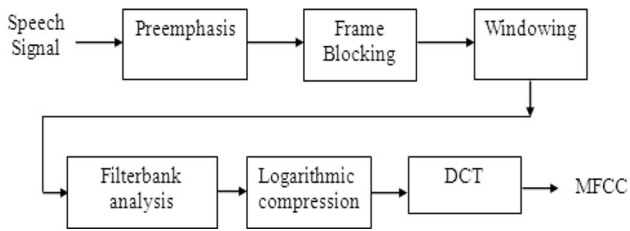


Fig. 4 Linear mel-filter bank channel outputs (MELSPEC) feature extraction process



**Fig. 5** Mel frequency cepstral coefficients (MFCC) feature extraction process

transform (DCT) on each of the windowed frames. The MFCC feature extraction process can be shown as in Fig. 5.

## 2.5 Hidden Markov model (HMM)

Some of the speech recognition task in recent decades has been done using the hidden Markov models (HMM). HMM can be defined as a stochastic process that produces output as a sequence of observations as in (Rabiner 1989; Rabiner and Juang 1993). It is extensively used in automatic speech recognition.

In HMM, the probabilistic parameters are the hidden states, observations, probability of the start state, transition and emission probabilities. In the present study, words are the hidden state sequence and acoustic signals are the observation sequence. During HMM training, the acoustic feature vectors are assigned to HMM state for each word and re-estimation of these HMM parameters are performed again and again to obtain the desired optimal values for each word. After models are trained for every word, these models can be tested using similar word utterances by different or same speakers to get the accuracy of the recognizer. The most likelihood calculation is done for all the models, and the one with the highest likelihood is selected.

## 2.6 Hidden Markov model toolkit (HTK)

One of the common toolbox for constructing HMM models in speech recognition is hidden Markov model toolkit (HTK). HTK was developed in Cambridge University Engineering Department (CUED). It is used for developing applications in various research areas but is primarily used for research in speech recognition areas. HTK consists of library modules and tools written in C (Evermann et al. 1997; Moreau 2002). In this present study, we have used the HTK tools for data preparation, feature extraction, model generation and testing.

## 3 Related work

This section represents the literature on some of the recent speech recognition works on languages that are prevalent in India.

In (Al-Qatab and Aïnon 2010), an Arabic automatic speech recognition engine that can recognize both continuous speech and isolated words have been proposed using the hidden Markov model toolkit and mel frequency cepstral coefficients (MFCC) as feature vectors. The system was trained based on tri-phones using Hidden Markov Model and using ten speakers. Testing was done using three speakers and maximum accuracy of 98.01 % was achieved.

A speaker-independent Arabic continuous speech recognition system was developed using both Sphinx and HTK tools in (Abushariah et al. 2010). The system uses mel frequency cepstral coefficients (MFCC) for extracting and was trained on seven hours of Arabic speech. The system obtained a word recognition accuracy of 93.88 %.

Another speaker-independent continuous automatic Arabic speech recognition system which consists of a total of 415 sentences spoken by 40 speakers was developed in (Abushariah et al. 2012). For the development of the system, both Sphinx and HTK tools have been used and a word recognition accuracy of 90.23 % was achieved.

In this linear predictive cepstral coefficients (LPCC) and mel frequency cepstral coefficients (MFCC) were used as features and their performances have been evaluated for Assamese phonemes using a multilayer perception network by (Bhattacharjee 2013). LPCC gives recognition accuracy 94.23 % compared to 89.14 % for MFCC when different speakers are tested.

A gender independent Bangla automatic speech recognition system has been proposed in (Hassan et al. 2012) which considers both male and female genders. The system was developed using medium sized Bangla vocabulary and gives an accuracy of 87.30 %.

The development of Swaranjali, an isolated word recognizer for digits in Hindi language has been shown in (Pruthi et al. 2000). The system uses linear prediction coding (LPC) and vector quantization (VQ) for feature extraction and the recognition of the system is done using HMM. The system gave a maximum accuracy of 97.14 % for the isolated word ‘teen’ representing the digit 3.

A Hindi speech recognition system has been developed using HTK in (Kumar and Aggarwal 2011). The system recognizes the isolated words using acoustic word model and is trained for 30 Hindi words. Eight speakers have been used for training and five speakers for testing to get an accuracy of 94.63 %.

Another connected-words speech recognition system for Hindi language is developed in (Kumar et al. 2012). The system uses MFCC as feature and used 12 speakers for recognizing a vocabulary of 102 words. The system has the overall word-accuracy of 87.01.

Mohamed and Nair (2012) in their paper have described a continuous Malayalam speech recognition system which

combines Hidden Markov Models (HMM) with Artificial Neural Networks (ANN) for a speech corpus consisting of 108 sentences with 540 words and a total of 3060 phonemes giving a performance of 86.67 % for word recognition.

A Marathi speaker dependent, isolated words speech recognition system for both offline and online speakers has been developed in (Shinde and Gandhe 2013). MFCC and DTW have been combined for word recognition. The system gives a maximum accuracy of 100 % for offline and 72.22 % for online speaker dependent word recognition.

Mehta et al. (2013) in their paper compared the performances of MFCC and LPC features using VQ for vowels and consonants in Marathi language. The system was trained for 48 words using two speakers. The MFCC feature was found to give on an average 99 % recognition accuracy compared to 77 % recognition accuracy for LPCC feature for both the speakers.

A Punjabi isolated words automatic speech recognition system using HTK is shown in (Dua et al. 2012). The system has been trained using eight speakers for 115 distinct Punjabi words. Mel Frequency Cepstral Coefficient (MFCC) and Hidden Markov Model (HMM) is used for training the isolated words. The system achieves maximum accuracy of 95.63 %.

Sigappi and Palanivel (2012) in their paper developed a speech recognition system for words spoken in Tamil language [18]. Mel Frequency Cepstral Coefficient (MFCC) technique is used to extract the features and the models chosen for the task are hidden Markov models

(HMM) and auto-associative neural networks (AANN). HMM yields a recognition rate of 95.0 %.

Another speech recognition system for Tamil language has been showed in (Krishna et al. 2014). MFCC and Integrated Phoneme Subspace (IPS) method are used to extract and Hidden Markov Model (HMM) is used for modeling. MFCC gives 74.67 % accuracy whereas IPS gives 84.00 % word accuracy.

Mankala et al. (2014) provides an isolated word speech recognition system using Hidden Markov Model Toolkit (HTK) for Telugu language. The system uses nine speakers to train the system for 113 Telugu words. The overall accuracy of the system using 10 HMM is 96.64 %.

Another isolated word speech recognition system for Telugu language is developed in (Bhaskar and Rao 2014). This system uses MFCC for feature extraction and HMM for model creation using Sphinx4. The vocabulary size for this system is two hundred and fifty Telugu words and testing is done by 10 speakers to get an accuracy of 91 %.

Table 1 depicts the summary of some of speech recognition accuracy with different features for some Indian languages.

#### 4 Assamese language

Assamese is one of the constitutionally recognized languages in India. It is mostly spoken in the North-East Indian state of Assam. Sanskrit, which is the mother script

**Table 1** Speech recognition rate of some Indian languages

Author	Language	Year	Features	Recognition rate (%)
Al-Qatab et al.	Arabic	2010	MFCC	98.01
Abushariah et al.	Arabic	2010	MFCC	93.88
Abushariah et al.	Arabic	2010	MFCC	90.23
Bhattacharjee	Assamese	2013	LPCC, MFCC	94.23 (LPCC) 89.14 (MFCC)
Hassan et al.	Bangla	2012	MFCC	87.3
Pruthi et al.	Hindi	2000	LPC	97.14
Kumar et al.	Hindi	2011	MFCC	94.63
Kumar et al.	Hindi	2012	MFCC	87.01
Mohamed and Nair	Malyalam	2012	MFCC	86.67
Shinde and Gandhe	Marathi	2013	MFCC, DTW	100 (offline) 72.22 (online)
Mehta et al.	Marathi	2013	MFCC, LPCC	99 (MFCC) 77 (LPCC)
Dua et al.	Punjabi	2012	MFCC	95.63
Sigappi et al.	Tamil	2012	MFCC	95.0
Krishna et al.	Tamil	2014	MFCC	74.67
Mankala et al.	Telugu	2014	MFCC	96.64
Bhaskar and Rao	Telugu	2014	MFCC	91

of many Indian languages like Hindi, Bengali etc. is also the mother of Assamese language. Besides Assam this language is also spoken in the neighboring states of Assam. In the present study, we have considered the vocabulary of Assamese digits from 0 to 9 as shown in Table 2.

### 5.2 System architecture

The architecture of the recognizer is mainly divided into four components, namely data preparation, feature extraction, training of word models, and testing as depicted in Fig. 6.

## 5 Experimental setup

### 5.1 System description

The recognition task is implemented for word models using HTK version 3.4.1 in Windows 7 operating system.

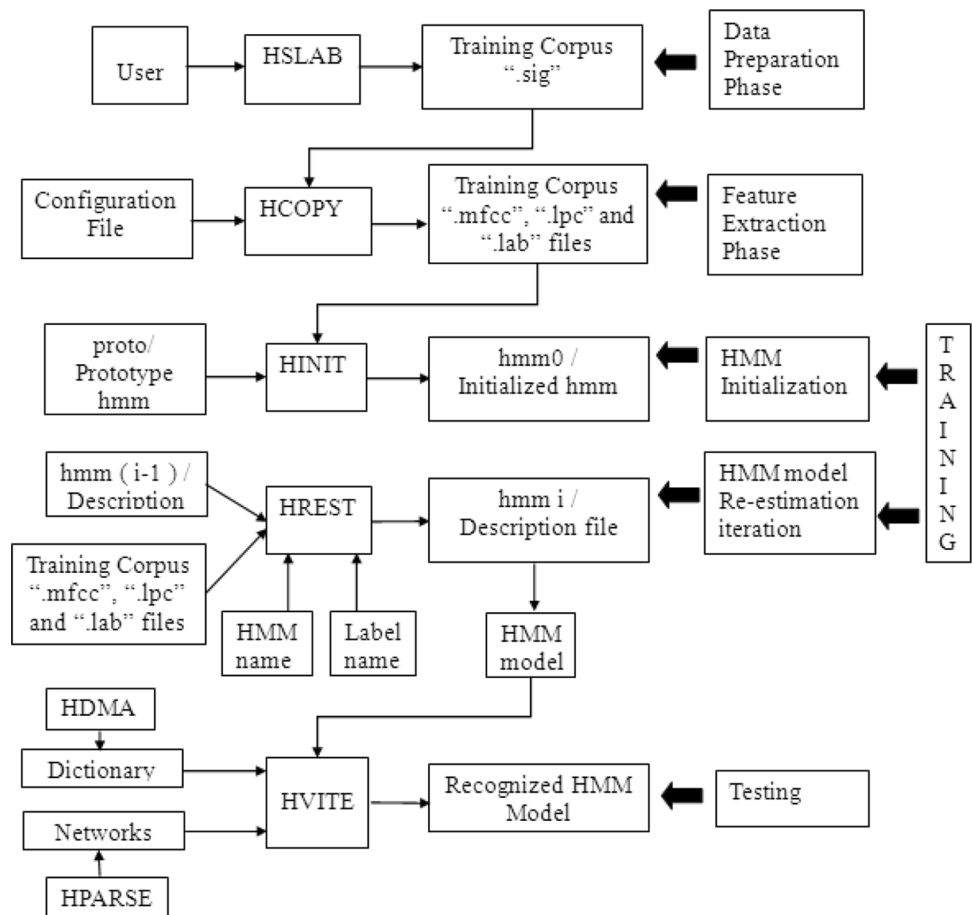
#### 5.2.1 Data preparation

In this phase recording and labeling of the speech signal is done. Speech recording is done in digital domain using dynamic microphone for 10 words corresponding to the digits 0 to 9. The recognizer is trained for 10 words each word corresponding to Assamese digit.

**Table 2** Assamese digits from 0 to 9 with their pronunciation

Word	0	1	2	3	4	5	6	7	8	9
Digit in English	Zero	One	Two	Three	Four	Five	Six	Seven	Eight	Nine
Digit in Assamese	“shounya”	“ek”	“dui”	“tini”	“chari”	“pas”	“choie”	“shaat”	“aath”	“no”

**Fig. 6** Architecture of the recognizer



Fifteen speakers, including seven male and eight female speakers were used to record the ten words. Each speaker recorded each of the ten words 10 times to give 100 samples per speaker. Thus a total of 1500 ( $15 \times 10 \times 10$ ) samples were collected. The speech signal is stored in.sig format and the labels are stored in.lab format.

The data recording is carried out at room environment with 16 kHz sampling rate and contains some amount of environmental noise. We have considered two categories of data in this research work noisy data, containing the environmental noise and clean data, without any environmental noise. For clean data, the environmental noise is removed using the noise removal tool available in Audacity.

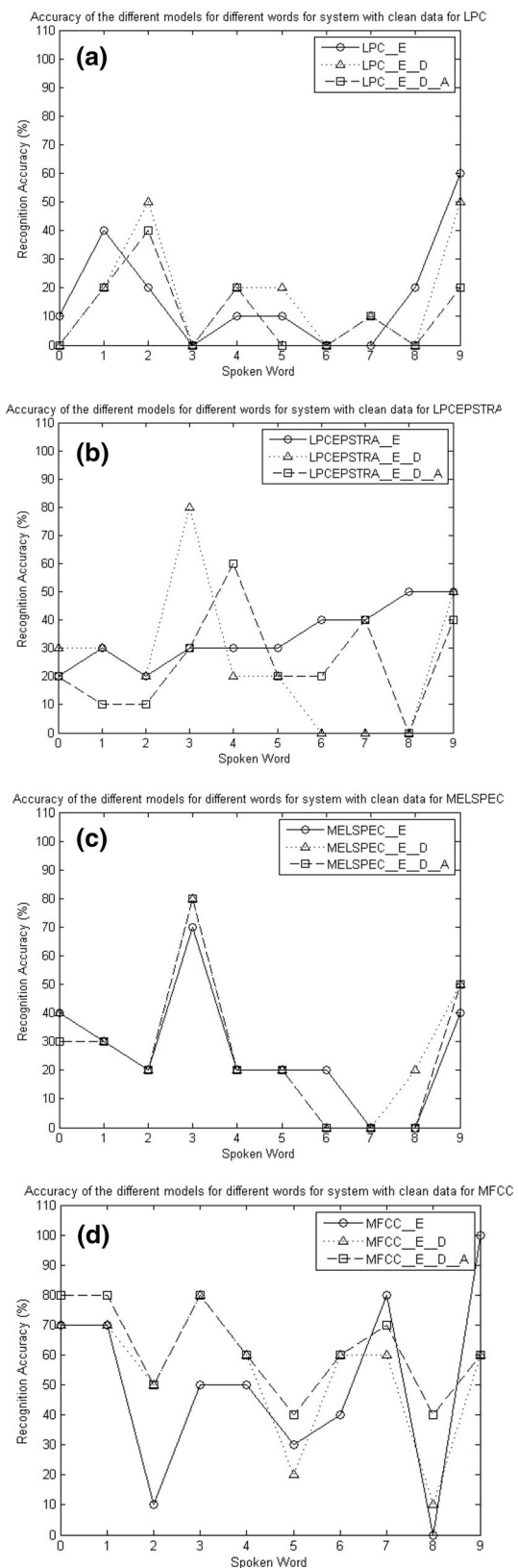
### 5.2.2 Feature extraction

Speech recognition tools cannot process on the speech files directly so it is required to convert to some type of parametric representation by extracting features from the speech files. In the present study linear predictive coding (LPC) analysis, LPC cepstral coefficients (LPCEPSTRA), linear mel-filter bank channel outputs (MELSPEC) and mel frequency cepstral coefficients (MFCC) are used as feature vectors. We have used three variants of all the above four features based on the number of features extracted from a speech signal. The feature extraction is done for both the categories of data.

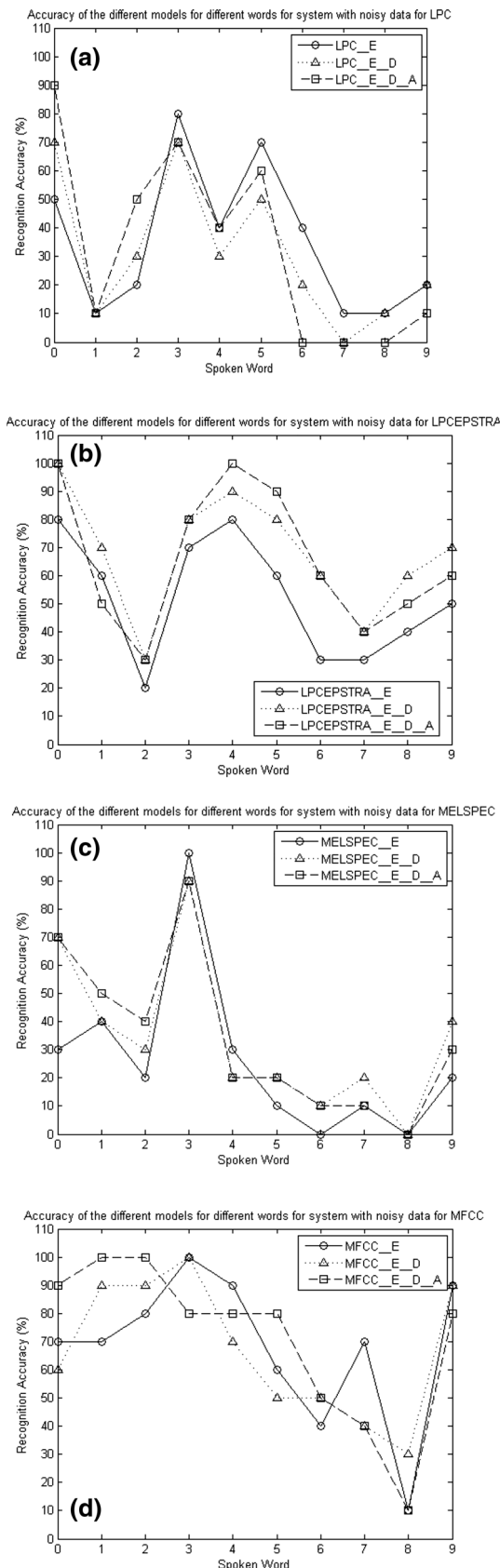
For LPC, we have considered LPC\_E as the first variant, which consists of 13 values where the first 12 are LPC coefficients along with \_E which represents the total energy in the particular speech frame. The second feature variant is LPC\_E\_D where the earlier 13 coefficients found in LPC\_E and their first order derivatives known as “Delta coefficients” are considered. The third variant feature is LPC\_E\_D\_A where the earlier 26 coefficients found in LPC\_E\_D and their second order derivatives known as “Acceleration coefficients” are considered per HMM model generation.

Similarly for LPCEPSTRA, we have considered LPCEPSTRA\_E as the first variant, which consists of 13 values where the first 12 are LPCEPSTRA coefficients along with \_E which represents the total energy in the particular speech frame. The second feature variant is LPCEPSTRA\_E\_D where the earlier 13 coefficients found in LPCEPSTRA\_E and their first order derivatives known as “Delta coefficients” are considered. The third variant feature is LPCEPSTRA\_E\_D\_A where the earlier 26 coefficients found in LPCEPSTRA\_E\_D and their second order derivatives known as “Acceleration coefficients” are considered.

For MELSPEC, we have considered MELSPEC\_0 as the first variant, which consists of 21 values where the first



**Fig. 7** Recognition rate for Assamese spoken digits for system with clean data for **a** LPC, **b** LPCEPSTRA, **c** MELSPEC, **d** MFCC



**Fig. 8** Recognition rate for Assamese spoken digits for system with noisy data for **a** LPC, **b** LPCEPSTRA, **c** MELSPEC, **d** MFCC

20 are MELSPEC coefficients along with MELSPEC coefficient c0 which represents the total energy in the particular speech frame. The second feature variant is MELSPEC\_E\_D where along with the earlier 21 coefficients found in MELSPEC\_0 and their first order derivatives known as “Delta coefficients” are considered. The third variant feature is represented as MELSPEC\_E\_D\_A where the earlier 42 coefficients found in MELSPEC\_E\_D and their second order derivatives known as “Acceleration coefficients” are considered.

Similarly, for MFCC, we have considered MFCC\_0 as the first variant, which consists of 13 values where the first 12 are MFCC coefficients along with MFCC coefficient c0 which represents the total energy in the particular speech frame. The second feature variant is MFCC\_0\_D where the earlier 13 coefficients found in MFCC\_0 and their first order derivatives known as “Delta coefficients” are considered. The third variant feature is MFCC\_0\_D\_A where the earlier 26 coefficients found in MFCC\_0\_D and their second order derivatives known as “Acceleration coefficients” are considered.

We use the HTK tool to get the MFCC and LPC vectors for each speech signal in both the training and testing phase. A configuration file is used which defines the configuration parameters.

### 5.2.3 Training of word

In the present study, we have considered three variants for each feature i.e. LPC, LPCEPSTRA, MELSPEC and MFCC which gives us a total of twelve different feature parameter set. For each of these above twelve parameters, LPC\_E, LPC\_E\_D, LPC\_E\_D\_A, LPCEPSTRA\_E, LPCEPSTRA\_E\_D, LPCEPSTRA\_E\_D\_A, MELSPEC\_0, MELSPEC\_E\_D, MELSPEC\_E\_D\_A, MFCC\_0, MFCC\_0\_D and MFCC\_0\_D\_A we have considered 5 different models for each parameter based on the number of hidden states giving us a total of  $(12 \times 5 = 60)$  sixty different acoustic models for any particular word in a particular category of data. Considering both the noisy and clean data we have a total of  $(60 \times 2 = 120)$  one hundred and twenty models for any word. The generated model is then re-estimated using the same speech signal to get the optimum values. These models are taken as reference models for testing unknown word files to get the most matching word.

In the first step, random initialization is done for each word in the vocabulary and a prototype is created. Each prototype has the same initialization and consists of non-emitting states representing the initial and last states of the model along with different numbers of hidden states between them. Different prototypes are created with different hidden states. The prototype file is created consisting

of mean and variance vectors and is used for initialization along with the lab files.

The next step consists of re-estimating the HMM parameters repetitively to obtain the desired optimal values for the HMM by using the Viterbi algorithm. Each word model is then re-estimated three times and we generate three HMM's per word in the dictionary. Taking the third re-estimated model as final model for that particular case of hidden numbers and parameter we have ( $3 \times 5 = 15$ ) fifteen such final models for all LPC, LPCEPSTRA, MELSPEC and MFCC feature parameters for every word in a particular category of data. So, a total of ( $12 \times 5 = 60$ ) sixty such models for each word which varies on the feature parameter, the number of feature values and hidden states considered in one category of data. Considering both the categories of clean and noisy data we have ( $60 \times 2 = 120$ ) one hundred and twenty models for each word. Thus considering the ten Assamese words, we have developed ( $120 \times 10 = 1200$ ) one thousand two hundred models in the present study.

#### 5.2.4 Task definition

The task grammar, a text file consists of recognition rules for the recognizer. The dictionary file is also created which is a text file and is used to develop an association among the task grammar and HMM. The grammar file is then compiled to generate the task network file.

#### 5.2.5 System testing

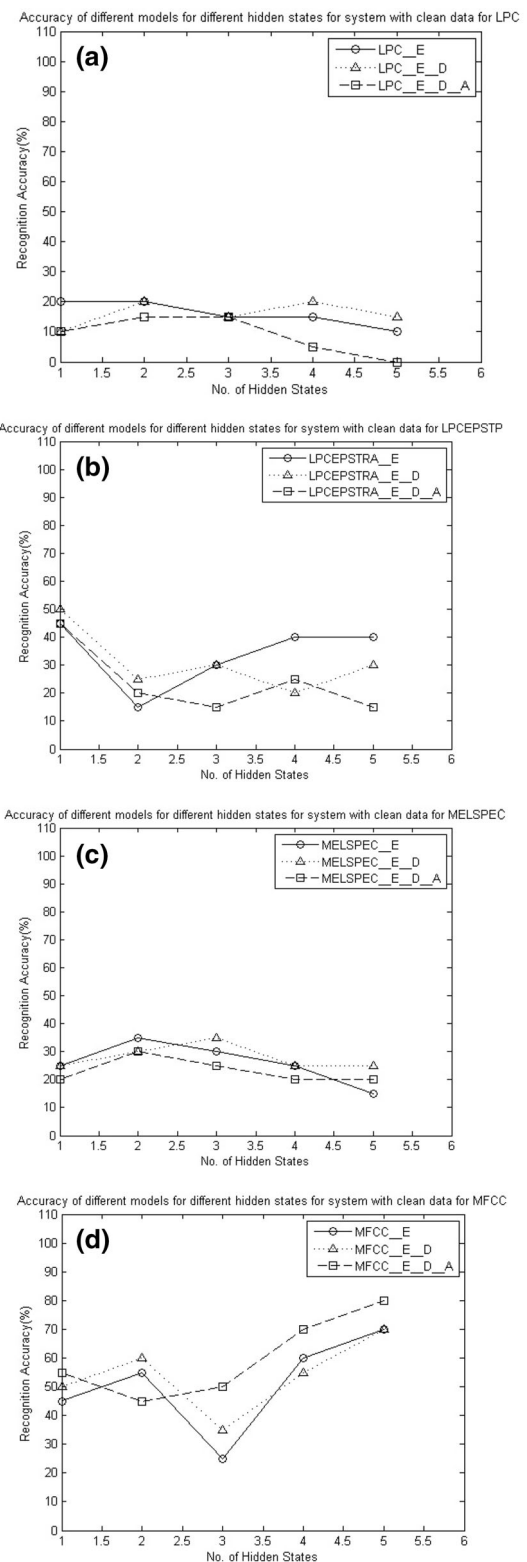
The testing of any unknown word is done in this phase. Three speakers were used to record all the words for testing. The test speech files are first converted to a series of acoustical feature vectors. The HTK tool Hvite is then used with the newly generated test mfcc file along with task grammar, task network and the previously generated HMMs to get the output in \*.mlf file. The signal is then matched against the HMMs and an output file is produced.

#### 5.2.6 Performance analysis

The output file is compared with the actual reference file and the system performance is computed.

The word error rate (WER) of the system is calculated using the formula given in Eq. 1:

$$\text{Word Error Rate (WER)} = 100 \% - \text{Percentage Accuracy} \quad (1)$$



**Fig. 9** Recognition rate for different hidden states for system with clean data for **a** LPC, **b** LPCEPSTRA, **c** MELSPEC, **d** MFCC



### 6 Results and discussion

The graphical representation of the recognition efficiency for 10(ten) Assamese spoken digits corresponding to male and female speakers are depicted in Figs. 7a–d and 8a–d. Similarly, graphical representation of recognition efficiency considering number of hidden states have been depicted in Figs. 9a–d and 10a–d.

It is evident from the graphs that digits 7 (pronounced as “shaat”) and 8 (pronounced as “aath”) have very low recognition rate. It is found that the MFCC feature has the highest recognition accuracy for clean data and LPCEPSTRA feature for noisy data. The performance based on recognition accuracy has been shown in the Tables 3, 4, 5 and 6.

In the present study, for the system developed using clean data the maximum accuracy received for LPC is 20 %, for LPCEPSTRA is 50 %, for MELSPEC is 35 % and for MFCC is 80 %. Also, for the system developed using noisy data the maximum accuracy received for LPC is 60 %, for LPCEPSTRA is 95 %, for MELSPEC is 60 % and for MFCC is 80 %. It is clearly observed that better performance in terms of recognition accuracy has been seen when the number of hidden states are increased. Taking into consideration all parameters, it has been found that HTK tool with 5 hidden states give the optimal performance depending upon the parameter used.

### 7 Conclusions and future work

An isolated word recognizer for Assamese language has been developed. We have followed the IPA standard for our system. The efficiency of the recognizer has been examined with different features. The recognizer gives a maximum accuracy of 80 % for MFCC for clean data and 95 % for LPCEPSTRA for noisy data. It is thus concluded in the present study that MFCC is a better parameter than LPC, LPCEPSTRA and MELSPEC for speech recognition for clean data and LPCEPSTRA is a better parameter than LPC, MELSPEC and MFCC for speech recognition for noisy data in Assamese language. Also for most of the words 100 % accuracy has been achieved for many features.

The future works involves improving the system by adding different noise environments and by increasing vocabulary size. Also, the work can be extended from an isolated word system to a connected-word system.

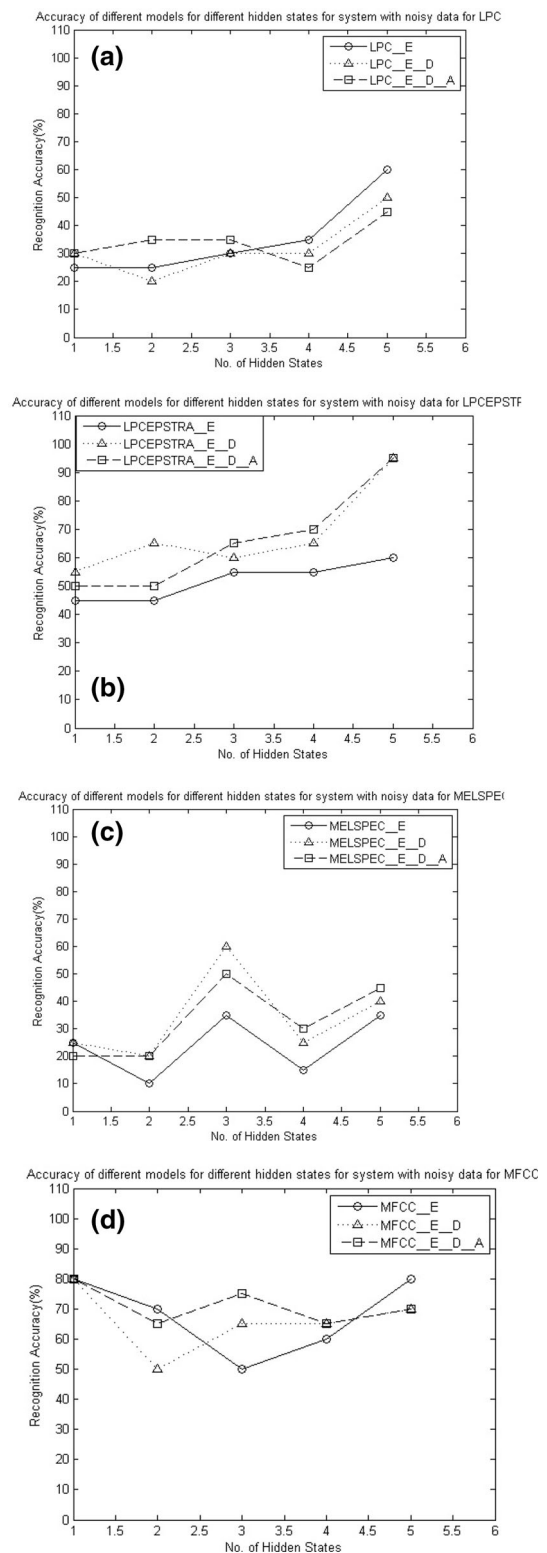


Fig. 10 Recognition rate for different hidden states for system with noisy data for a LPC, b LPCEPSTRA, c MELSPEC, d MFCC

**Table 3** Accuracy of the different models for different number of hidden states for system with clean data

Features	Recognition rate (%)							
	Number of hidden states					Average accuracy	Maximum accuracy	
	1	2	3	4	5			
LPC_E	20	20	15	15	10	16	20	
LPC_E_D	10	20	15	20	15	16	20	
LPC_E_D_A	10	15	15	05	0	09	15	
LPCEPSTRA_E	45	15	30	40	40	34	45	
LPCEPSTRA_E_D	50	25	30	20	30	31	50	
LPCEPSTRA_E_D_A	45	20	15	25	15	24	45	
MELSPEC_0	25	35	30	25	15	26	35	
MELSPEC_E_D	25	30	35	25	25	28	35	
MELSPEC_E_D_A	20	30	25	20	20	23	30	
MFCC_0	45	55	25	60	70	51	70	
MFCC_0_D	50	60	35	55	70	54	70	
MFCC_0_D_A	55	45	50	70	80	60	80	

**Table 4** Accuracy of the different models for different Assamese spoken digits for system with clean data

Features	Recognition rate (%)										Average accuracy	Maximum accuracy
	Zero	One	Two	Three	Four	Five	Six	Seven	Eight	Nine		
LPC_E	10	40	20	0	10	10	0	0	20	60	17	60
LPC_E_D	0	20	50	0	20	20	0	10	0	50	17	50
LPC_E_D_A	0	20	40	0	20	0	0	10	0	20	11	40
LPCEPSTRA_E	20	30	20	30	30	30	40	40	50	50	34	50
LPCEPSTRA_E_D	30	30	20	80	20	20	0	0	0	50	25	80
LPCEPSTRA_E_D_A	20	10	10	30	60	20	20	40	0	40	25	60
MELSPEC_0	40	30	20	70	20	20	20	0	0	40	26	70
MELSPEC_E_D	40	30	20	80	20	20	0	0	20	50	28	80
MELSPEC_E_D_A	30	30	20	80	20	20	0	0	0	50	25	80
MFCC_0	70	70	10	50	50	30	40	80	0	100	50	100
MFCC_0_D	70	70	50	80	60	20	60	60	10	60	54	80
MFCC_0_D_A	80	80	50	80	60	40	60	70	40	60	62	80

**Table 5** Accuracy of the different models for different number of hidden states for system with noisy data

Features	Recognition rate (%)							
	Number of hidden states					Average accuracy	Maximum accuracy	
	1	2	3	4	5			
LPC_E	25	25	30	35	60	35	60	
LPC_E_D	30	20	30	30	50	32	50	
LPC_E_D_A	30	35	35	25	45	34	45	
LPCEPSTRA_E	45	45	55	55	60	52	60	
LPCEPSTRA_E_D	55	65	60	65	95	68	95	
LPCEPSTRA_E_D_A	50	50	65	70	95	66	95	
MELSPEC_0	25	10	35	15	35	24	35	
MELSPEC_E_D	25	20	60	25	40	34	60	
MELSPEC_E_D_A	20	20	50	30	45	33	50	

**Table 5** continued

Features	Recognition rate (%)						
	Number of hidden states						
	1	2	3	4	5	Average accuracy	Maximum accuracy
MFCC_0	80	70	50	60	80	68	80
MFCC_0_D	80	50	65	65	70	66	80
MFCC_0_D_A	80	65	75	65	70	71	80

**Table 6** Accuracy of the different models for different Assamese spoken digits for system with noisy data

Features	Recognition rate (%)											Average accuracy	Maximum accuracy
	Zero	One	Two	Three	Four	Five	Six	Seven	Eight	Nine			
LPC_E	50	10	20	80	40	70	40	10	10	20	35	80	
LPC_E_D	70	10	30	70	30	50	20	0	10	20	33	70	
LPC_E_D_A	90	10	50	70	40	60	0	0	0	10	33	90	
LPCEPSTRA_E	80	60	20	70	80	60	30	30	40	50	52	80	
LPCEPSTRA_E_D	100	70	30	80	90	80	60	40	60	70	74	100	
LPCEPSTRA_E_D_A	100	50	30	80	100	90	60	40	50	60	66	100	
MELSPEC_0	30	40	20	100	30	10	0	10	0	20	26	100	
MELSPEC_E_D	70	40	30	90	20	20	10	20	0	40	34	90	
MELSPEC_E_D_A	70	50	40	90	20	20	10	10	0	30	34	90	
MFCC_0	70	70	80	100	90	60	40	70	10	90	8	100	
MFCC_0_D	60	90	90	100	70	50	50	40	30	90	67	100	
MFCC_0_D_A	90	100	100	80	80	80	50	40	10	80	63	100	

**References**

Abushariah, M. A., Aion, R. N., Zainuddin, R., Elshafei, M., & Khalifa, O. O. (2010). Natural speaker-independent Arabic speech recognition system based on Hidden Markov Models using Sphinx tools. In *Computer and Communication Engineering (ICCCE)* (pp. 1–6), 2010 International Conference on, IEEE.

Abushariah, M. A. A. M., Aion, R. N., Zainuddin, R., Elshafei, M., & Khalifa, O. O. (2012). Arabic speaker-independent continuous automatic speech recognition based on a phonetically rich and balanced speech corpus. *International Arab Journal of Information Technology*, 9(1), 84–93.

Al-Qatab, B. A., & Aion, R. N. (2010). Arabic speech recognition using hidden Markov model toolkit (HTK). In *Information Technology (ITSim)* (Vol. 2, pp. 557–562), 2010 International Symposium in, IEEE.

Bhaskar, P. V., & Rao, S. R. M. (2014). *Telugu Speech Recognition System development using MFCC based Hidden Markov Model technique with Sphinx-4*.

Bhattacharjee, U. (2013). A comparative study of LPCC and MFCC features for the recognition of assamese phonemes. In *International Journal of Engineering Research and Technology* (Vol. 2, No. 1 (January-2013)). ESRSA Publications.

Bourlard, H., & Morgan, N. (1998). Hybrid HMM/ANN systems for speech recognition: Overview and new research directions. In *Adaptive processing of sequences and data structures* (pp. 389–417). Berlin: Springer.

Dua, M., Aggarwal, R. K., Kadyan, V., & Dua, S. (2012). Punjabi automatic speech recognition using HTK. *IJCSI International Journal of Computer Science Issues*, 9(4), 1694-0814.

Eslam Mansour Mohammed, E. M. M., Mohammed Sharaf Sayed, M. S. S., Abdallaa Mohammed Mosehly, A. M. M., & Abdelaziz Alsayed Abdelnaiem, A. A. A. (2013). LPC and MFCC performance evaluation with artificial neural network for spoken language identification. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 6(3), 55–66.

Evermann, G., Gales, M., Hain, T., Kershaw, D., Liu, X., Moore, G., & Woodland, P. (1997). *The HTK book* (Vol. 2). Cambridge: Entropic Cambridge Research Laboratory.

Hassan, F., Kotwal, M. R. A., Khan, M. S. A., & Huda, M. N. (2012). Gender independent Bangla automatic speech recognition. In *Informatics, Electronics & Vision (ICIEV)* (pp. 144–148), 2012 International Conference on, IEEE.

Krishna, K. M., Lakshmi M. V., & Lakshmi, S. S. (2014). Feature extraction and dimensionality reduction using IPS for isolated tamil words speech recognizer. *International Journal of Advanced Research in Computer and Communication Engineering*, 3(3).

Kumar, K., & Aggarwal, R. K. (2011). Hindi speech recognition system using HTK. *International Journal of Computing and Business Research*, 2(2), 2229–6166.

Kumar, K., Aggarwal, R. K., & Jain, A. (2012). A Hindi speech recognition system for connected words using HTK. *International Journal of Computational, Systems Engineering*, 1(1), 25–32.

- Mankala, S. R., Bojja, S. R., Ramaiah, V. S., & Rao, R. R. (2014). Automatic speech processing using HTK for Telugu language. *International Journal of Advances in Engineering & Technology*, 6(6), 2572–2578.
- Mehta, L. R., Mahajan, S. P., & Dabhade, A. S. (2013). Comparative study of MFCC and LPC for Marathi isolated word recognition system. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 2(6), 2133–2139.
- Mohamed, A., & Nair, K. N. (2012). HMM/ANN hybrid model for continuous Malayalam speech recognition. *Procedia Engineering*, 30, 616–622.
- Moreau, N. (2002). HTK v. 3.1 Basic Tutorial. Technische Universität Berlin.
- Pruthi, T., Saksena, S., & Das, P. K. (2000). Swaranjali: Isolated word recognition for Hindi language using VQ and HMM. In *International Conference on Multimedia Processing and Systems (ICMPS)*, IIT Madras.
- Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2), 257–286.
- Rabiner, L. R., & Juang, B. H. (1993). *Fundamentals of speech recognition* (Vol. 14). Englewood Cliffs: PTR Prentice Hall.
- Shinde, M. B., & Gandhe, D. S. (2013). Speech processing for isolated Marathi word recognition using MFCC and DTW features. *International Journal of Innovations in Engineering and Technology*, 3(1).
- Sigappi, A. N., & Palanivel, S. (2012). Spoken word recognition strategy for Tamil language. *International Journal of Computer Science Issue*, 9(1), 1694–0814.