International Journal of Engineering and Technical Research (IJETR)
ISSN: 2321-0869, Volume-2, Issue-11, November 2014

Implementation of Automatic Segmentation of Speech Signal for Phonetic Engine in Malayalam

Deekshitha G, Jubin James Thennattil, Leena Mary

***Abstract*—Automatic speaker recognition has immense applications and is in research for several decades. Phonetic Engine (PE) is the first stage of Automatic Speech Recognition (ASR), that converts input speech into a sequence of phonetic symbols, which is to be succeeded by a language model to incorporate lexical knowledge of given language. Phonetic Engine uses pattern recognition approach by recognizing the phonemes present in acoustic signal. Due to the large number of phonemes in Malayalam language, phonemes classes become more confusable, and therefore performance of the developed phonetic engine seems inadequate. To improve the performance of the real time phonetic engine, we have developed a front-end for automatically segmenting long test utterances to short segments. This is done by detecting pauses automatically using a feed forward neural network designed for speech/non-speech classification. The phonetic engine with this segmentation front end performs better.**

***Index Term*—ANN Classifier, Automatic Speech Recognition, Phonetic engine, Prosody, Segmentation**

I. INTRODUCTION

Recognition of speech signal has immense applications and researchers are trying to build a state of art speech recognition engine over decades [1]. The system which converts speech signal to text is termed as Automatic Speech Recognition (ASR) system. ASR is usually built in two stages. Phonetic engine is the first stage of ASR and it converts speech signal to phonetic symbols. Phonetic engine uses the acoustic phonetic information present in the speech signal in terms of features such as Mel Frequency Cepstral Coefficients (MFCC). The phonetic engine is followed by a language model to incorporate lexical knowledge of given language in ASR. Figure 1 shows the block diagram of an ASR.

Malayalam language is spoken mostly by people of Kerala and Lakshadweep. It is one of the scheduled languages of India, which also has a classical language status. Implementing an automatic speech recognition engine in Malayalam has got much significance in cultural, economical domain. Malayalam language consists of fifteen vowels, forty one consonants and six special phonemes called 'chillu'. Considering these sixty two alphabets, we have considered frequently occurring forty phonemes which are necessary for creation of phonemic classes in Malayalam.

Manuscript received November 22, 2014.

Deekshitha G, Electronics and Communication, Rajiv Gandhi Institute of Technology, Kottayam, Kerala, India.

Jubin James Thennattil, Electronics and Communication, Rajiv Gandhi Institute of Technology, Kottayam, Kerala, India.

Leena Mary, Electronics and Communication, Rajiv Gandhi Institute of Technology, Kottayam, Kerala, India

There are many issues in creating a real time, large vocabulary phonetic engine for continuous speech. The speech signal may not be recorded at studio environment and the silence region may contain some noise/energy regions due to background disturbances. Phonemes like plosives and fricatives, have lower energy compared to vowels and may be misclassified as silence/ background noise. Hence there are insertions, substitution errors in phonetic engine due to the testing environment and sufficiently large number of classes.

Prosody is of interest to ASR researchers, as it important for human speech recognition [2, 3]. In all languages, prosody is used to convey structural, semantic, and functional information. Prosody provides valuable information, often not available from text alone; for example, information on phrasing and disfluencies, emotion etc. A crucial step toward robust information extraction from speech is the automatic determination of topic, sentence, and phrase boundaries. Such locations are obvious in text (via punctuation, capitalization, formatting) but are absent or hidden in speech output [2]. Prosody in terms of long pauses is useful to humans for parsing longer utterances to shorter ones. This has motivated us to use prosodic characteristic like pause for automatically segmenting test utterances to shorter phrases. This segmentation helps to decreases some misclassification of silence to other phonemes.

The paper is organized as follows: Section II describes the baseline phonetic engine and the issues faced. Modified phonetic engine is explained in Section III. Descriptions about automatic segmentation are given in Subsection III.A. The performance of the proposed system is evaluated in Section IV. Finally the paper is wrapped up with a conclusion and scope for future work in Section V.

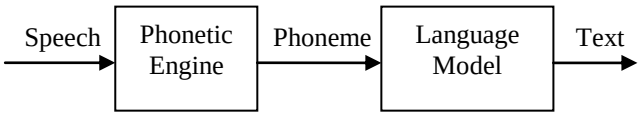

Figure 1: Block schematic of an ASR

II. IMPLEMENTATION OF PHONETIC ENGINE

Automatic speech recognition consists of transformation of the speech signal into a sequence of symbols corresponding to the sub word units of speech, and conversion of the symbol sequence into a text. Typically, continuous speech recognition is performed in the following steps: (1) speech signal-to-symbol (phonetic/syllabic) transformation, and (2) symbol-to-text conversion. Speech signal-to-symbol transformation is performed by a phonetic engine as shown in

Figure 2, and symbol-to-text conversion by imposing phonemic/lexical knowledge of the language.

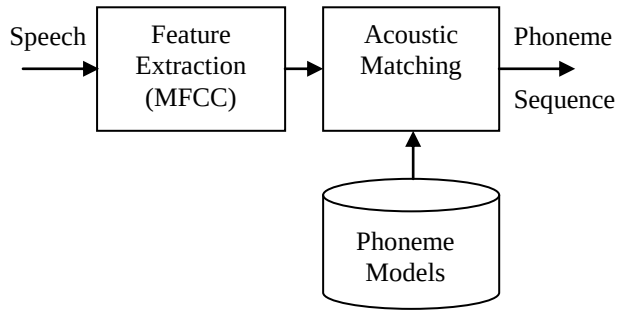

Figure 2: Block schematic of baseline phonetic engine

Database for implementing phonetic engine (PE) is collected from different regions of Kerala. The collected data consists of Malayalam speech spoken by people of different age group, gender etc. Data is collected in three different modes, namely read speech, extempore speech and conversational speech. There are many differences in each mode and hence we have developed three different phonetic engines. Read speech data was mostly taken from All India Radio (AIR).

At the next step, this database was manually transcribed at the phoneme level. Manual transcription is done using International Phonetic Alphabet (IPA) symbols [4]. Analyzing the IPA transcription of the database, it was found that there are large numbers of IPA symbols whose number of occurrences are very small in the data. This will lead to less number of examples for training from a limited transcribed data. So such less frequently occurring symbols are mapped to closely resembling IPA symbol that has similar production and perception characteristics. Thus we have selected 40 classes of phonemes including silence for the implementation of phonetic engine.

Mel frequency cepstral analysis of speech is based on human perception. Mel-Frequency Cepstrum (MFC) is a representation of the short-term power spectrum of a speech, based on a linear cosine transform of a log power spectrum on a nonlinear Mel scale of frequency. Mel Frequency Cepstral Coefficients are used to represent this cepstrum. For implementing phonetic engine, we have used 39 dimensional features that include MFCCs (12 dimensional), energy (1 dimensional), their first order derivative coefficients (13 dimensional) and second order derivative coefficients (13 dimensional). Features are generated for each frame of speech. These features are used for training and testing of phoneme models.

We used Hidden Markov Models (HMM) for building phoneme models, using HMM Tool Kit (HTK) [5]. It is a general purpose toolkit but optimized for speech recognition. We have used 75 % of read data for training and rest for testing. We have developed phonetic engine using 40 phoneme models as well as its real time interface. In the testing phase, features that are extracted by the feature extraction module is compared against a set of phoneme models as shown in Figure 2, to identify the sound that was produced. A screenshot of the real time interface is shown in Figure 3. We have incorporated real time live recording facility, loading, saving, and display of speech waveform with this real-time PE interface.

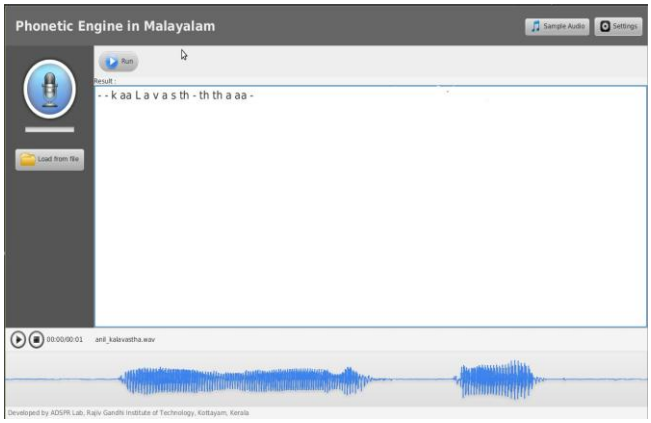

Figure 3: Screenshot of PE while testing word '*kalavastha*'

Upon analyzing the performance of the baseline phonetic engine, it is observed that there are many insertion errors while testing with long test utterances. But the engine works better for short utterances and has better accuracy due to the removal of long silence in middle of data, thus avoiding the possible mismatch with low energy phonemes. Hence, we are motivated to use a prosodic characteristic like pause to improve the performance of the baseline model, which is described in the next section.

III. Modified Phonetic Engine

In order to improve the performance of phonetic engine for long speech, we added an automatic phrase-like segmentation unit as a front-end to the existing baseline phonetic engine. The block schematic of the modified phonetic engine is shown in Figure 4. An Artificial Neural Network (ANN) based classifier system was developed to classify the input speech utterance into speech or non-speech segments which then helps in segmenting the input speech. The automatic segmentation unit accepts the longer test speech as input and produces a set of short segments which are then applied to the baseline phonetic engine. Since the phonetic engine performs well with short utterances, the engine's accuracy got improved.

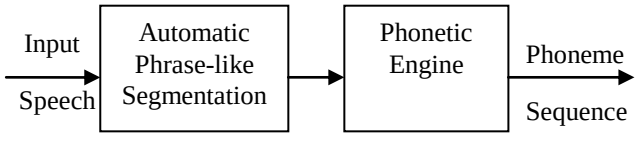

Figure 4: Block schematic of modified PE

A. *Automatic Phrase-like Segmentation*

Prosodic knowledge [6] like pause is used to automatically segment the long speech utterance to short phrases. Figure 5 shows the basic block diagram of automatic segmentation. The relevant features are extracted from speech signal, which are then fed to ANN classifier. Each frame is classified as speech or non speech. The speech signal is sliced if there are long pauses based on the speed of utterance.

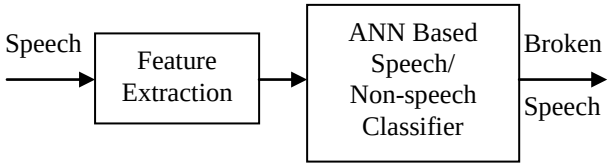

Figure 5: Block diagram for automatic segmentation

Prosodic knowledge [6] like pause is used to automatically segment the long speech utterance to short phrases. Figure 5 shows the basic block diagram of automatic segmentation. The relevant features are extracted from speech signal, which are then fed to ANN classifier. Each frame is classified as speech or non speech. The speech signal is sliced if there are long pauses based on the speed of utterance.

1) Features for Speech/Non-speech Classification

As a part of the preliminary analysis, some discriminative features [7] helpful for speech, non-speech classifications were identified. Those features include Short Time Energy (STE), Spectral Flatness Measure (SFM), voicing information and Most Dominant Frequency (MDF) . With these features, the classifier shows better performance in classifying the frames into speech/non-speech. In this work, the input signal is windowed into 20ms with 10 ms overlap in order to extract frame level features as shown in Figure 6.

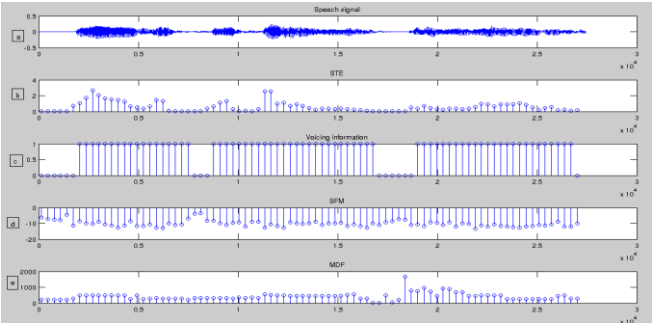

Figure 6: (a) Speech signal, (b) STE, (c) voicing information, (d) SFM and (e) MDF

- Short Time Energy (STE)

One of the basic short-time analysis functions useful for speech signals is the short time energy. The STE associated with non-speech regions is relatively smaller compared to speech regions. Hence STE is useful for speech/non-speech classification.

$$E(n) = \sum_{m=-\infty}^{\infty} [S(m)]^2 w(n-m) \qquad (1)$$

where *E(n)* is the STE of nth frame, *S(m)* is the speech signal and *w(n)* is the window. Figure 6(b) shows the STE of given speech segment.

- Voicing Information

This data is obtained by calculating the pitch of the input speech signal. It shows '0'for unvoiced segments and '1'for voiced segments. Figure 6(c) shows the plot of voicing information.

- Spectral Flatness Measure (SFM)

It is a measure of the noisiness of spectrum and is a good feature in voiced/unvoiced/silence detection.

$$SFM = 10 \log_{10} \frac{GM}{AM} \qquad (2)$$

Where *AM* and *GM* are arithmetic and geometric means of speech spectrum respectively. Figure 6(d) shows the SFM of a speech signal.

- Most Dominant Frequency (MDF)

MDF component is computed by finding the frequency corresponding to the maximum value of the spectrum magnitude. Figure 6(e) shows the plot of the MDF corresponding to a speech waveform given in Figure 6(a).

2) Speech/Non-speech Classifier System

As explained, the feature vectors for each short frame are calculated and normalized. Then these features are applied to an ANN based classifier for speech/non-speech classification. Here a multilayer feed forward Artificial Neural Network (ANN) [8, 9] is used as a classifier.

A feed-forward back propagation network is created with five layers: one input layer, one output layer and three hidden layers [9]. The final classifier has a structure 4L 8N 4N 2N 1L where L represents linear neurons and N represents non-linear neurons. Here the non-linear neurons use 'log sigmoid activation function. Feed forward networks have one-way connections from input to output layers.

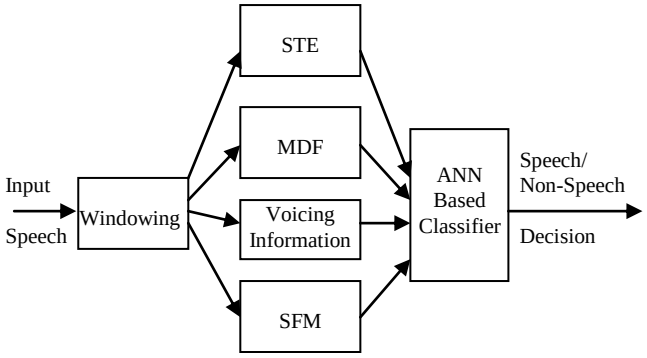

Figure 7: ANN based classifier for speech/non-speech classification

For each frame, the normalized feature vectors as well as the labeled output vector were given to train the ANN classifier. The ANN classifier then separates each frame into speech/non-speech segments. Figure 7 demonstrates the working of ANN classifier. If there are large numbers of consecutive non-speech frames, it is marked as a long pause. According to the speed of the input utterance, a limit is made for fixing the duration of long pause for segmentation. Depending upon this limit, the segmentation is done.

IV. Experimental Results

A. Automatic Segmentation

For testing, some long read speeches were collected. The speech data is segmented into frames of 20ms with 10ms overlap. The 4 feature vectors (STE, MDF, SFM, voicing information) of each frame are calculated and applied to the network classifier, so that the network outputs are interpreted as a label (S/N) that suits the best for the given feature vector.

Then the classifier output is smoothed using a median filter. A limit is fixed for determining the duration of long pauses based on the speed of the input speech. Test speech is segmented at the long pauses .and then applied to the phonetic engine. The ANN designed for speech/non-speech classification performs well. The accuracy in determining of speech/ non- speech segments are shown in the Table 1.

Table 1: Frame level percentage accuracy of speech/non-speech classification

Speech Data	Actual Output	Smoothed Output
Test 1	84.04	96.95
Test 2	85.20	93.27
Test3	93.78	95.56

B. Phonetic Engine

Before adding the automatic segmentation unit, evaluation of read mode phonetic engine was done by using 1 hour 15 minutes of data (around one Lakh of phoneme units) as shown in Table 2 for different number of Gaussian mixtures. The insertion losses, deletion are evaluated. It has been noticed that, the engine gives out better transcription results for small input data. But for long input utterances it gives out increased errors in transcription. Insertion errors also seem to be very high.

Table 2: Performance of baseline PE in Read mode

No. of mixtures	% Correctness
16	38.99
32	40.61
64	42.57

For other modes, the result is bad compared to read mode due to background noises, long pauses etc.. By integrating the automatic segmentation unit with the phonetic engine, the performance of the engine improved. The performance of the modified PE was evaluated by giving the segmented inputs to the phonetic engine, by incorporating automatic segmentation as a front end. By doing this, an improvement in accuracy is obtained. Improvement of 10 - 20 % is observed according to the mode and silence content of test data.

V. Conclusion and Future Work

An artificial neural network based frame level classifier is created for speech/non-speech classification. A real time large vocabulary phonetic engine in Malayalam is created. Accuracy is improved for long test speech utterances by incorporating phrase like segmentation of test utterance s front end to phonetic engine. This engine works in real time without need of complex studio environment and can be incorporated in many day to day applications. Future work includes incorporation of language model to create the performance of a commercial ASR in Malayalam language.

Acknowledgment

The authors would like to thank Department of Electronics and Information Technology, Government of India for providing financial assistance for the work discussed in the paper.

References

[1] Sadaoki Furui. 2006. 50 Years of Progress in Speech and Speaker Recognition Research. ECTI Transactions on Computer and Information Technology, vol 1, No. 2: 64-71.

[2] E. Shriberg and A. Stolcke. 2004. Direct Modeling of Prosody: An Overview of Applications in Automatic Speech Processing. Proc. ISCA Int. Conf. Speech Prosody, 2004.

[3] Leena Mary. 2012. Extraction and Representation of Prosody for Speech and Language Recognition. Springer Briefs in Electrical and Computer Engineering,, http://www.springer.com/978-1-46141158-1.

[4] Sreejith A, Leena Mary, Riyas K S, Joseph A and Augustine A. 2013. Automatic prosodic labeling and broad class Phonetic Engine for Malayalam. Control Communication and Computing (ICCC), 2013 International Conference on, IEEE, 522-526.

[5] Giampiero Salvi. 2003. HTK Tutorial. KTH, Dept. of Speech, Music and Hearing, Drottning Kristinas v.31,Stockholm, Sweden.

[6] E. Shriberg, A. Stolcke, D. Hakkani-Tur, and G. Tur. 2000. Prosody-based automatic segmentation of speech into sentences and topics, Speech Commun., pp.127 -154 sept.2000.

[7] Ginsha Elizabeth George, M.R Mahalakshmi and Leena Mary. 2011. Complementary Acoustic Features for Voice Activity Detection. Annual International Conference on Emerging Research Areas, AICERA April 2011 proceedings, Kottayam, India, ID:445, 329-333.

[8] Fu Guojiang. 2011. AA Novel Isolated Speech Recognition Method based on Neural Network. International Conference on Networking and Information Technology, IPCSIT vol.17 (2011) (2011) IACSIT Press, Singapore.

[9] Wouter Gevaert, Georgi Tsenov, and Valeri Mladenov. 2010. Neural Networks for Speech Recognition, Journal of Automatic Control, University of Belgrade, vol. 20:1-7.

[10] R. Kuhn and R. de Mori, "A cache-base natural language model for speech recognition", IEEE PAMI, vol. 12, pp. 570–583, June 1990.

Deekshitha G. graduated from Cochin University of Science and Technology in Electronics and Communication Engineering in 2012. She is currently doing Masters degree in Advanced Communication and Information Systems at Rajiv Gandhi Institute of Technology, Kottayam Kerala, India. Her areas of interest are speech processing and image processing.

Jubin James Thennattil received his B. Tech degree from Calicut University in Electronics and Communication Engineering and M. Tech degree specialized in Advanced Communication and Information Systems from M. G University, Kerala. He is currently working as Research Fellow with Advanced Digital Signal Processing Laboratory, Rajiv Gandhi Institute of Technology, Kottayam. His research areas include pattern recognition, signal processing for communication, and speech signal processing.

Leena Mary received her Bachelor's degree from Mangalore University in 1988. She obtained her MTech from Kerala University and Ph.D. from Indian Institute of Technology, Madras, India. She has 23 years of teaching experience. Currently she is working as Professor in Electronics and Communication Engineering at Rajiv Gandhi Institute of Technology, Kottayam, Kerala, India. Her research interests are speech processing, speaker forensics, signal processing and neural networks. She has published several research papers which include a book on Extraction and Representation of Prosody for Speaker, Speech and Language Recognition by Springer. She is a member of IEEE and a life member of Indian Society for Technical Education.