SPEECH RECOGNITION SYSTEM FOR TAMIL LANGUAGE USING CMUSPHINX

A.R.F.S. FANOON[134] & G.A.I. UWANTHIKA[135]

CORRESPONDENCE: FANOONARFS@GMAIL.COM

ABSTRACT

SPEECH RECOGNITION IS THE PROCESS OF CONVERTING AN AUDIO SIGNAL INTO SEQUENCES OF WORDS PAVING THE WAY FOR ENABLING A RICH HUMAN-COMPUTER INTERACTION IN MANY EMERGING APPLICATIONS. IT IS A VERY DIFFICULT AND A COMPLEX TASK TO RECOGNIZE SPEECH USING A COMPUTER THOUGH SEVERAL APPROACHES HAVE BEEN MADE TO DEVELOP AN ACCURATE SPEECH RECOGNITION SYSTEM. MOST OF THE RESEARCHES HAVE BEEN DONE FOR ENGLISH, CHINESE, MANDARIN AND ARABIC LANGUAGES WHILE A VERY FEW HAVE BEEN DONE FOR TAMIL LANGUAGE. THEREFORE, THIS RESEARCH INTENDS TO DEVELOP A CONTINUOUS SPEECH RECOGNITION SYSTEM FOR TAMIL LANGUAGE. TAMIL LANGUAGE HAS 12 VOWELS AND 18 CONSONANTS. THIS PAPER IS FOCUSED ON BUILDING A TAMIL SPEECH RECOGNITION SYSTEM USING CMUSPHINX TOOLKIT. THE SYSTEM WAS DEVELOPED FOR SEVERAL DESKTOP APPLICATIONS WITH THE AIM THAT IN THE FUTURE, THIS SYSTEM COULD BE INTEGRATED WITH THOSE APPLICATIONS AND EXECUTES THOSE APPLICATIONS THROUGH TAMIL SPEECH INPUT RATHER THAN TYPING AND CLICKIN. AUDIO RECORDINGS FROM SEVERAL PEOPLE WERE RECORDED IN DIFFERENT ENVIRONMENT WHICH WERE LATER CONVERTED TO WAV FILE FORMAT. A TEXT CORPUS, TRANSCRIPTION FILE AND FILEIDS FILE FOR BOTH TRAINING AND TESTING DATABASE WERE PREPARED ACCORDINGLY. THREE MODELS NAMELY ACOUSTIC MODEL, LANGUAGE MODEL AND LEXICON MODEL HAVE BEEN DEVELOPED. HIDDEN MARKOV MODEL WAS EMPLOYED FOR BUILDING ACOUSTIC MODEL. THE PERFORMANCE OF THE SYSTEM OVER VARIOUS SPEAKER SUBSETS OF DIFFERENT SEX, AGE AND DIALECT WAS EXAMINED. THE WORD ERROR RATE WAS CALCULATED IN ORDER TO MEASURE THE PERFORMANCE OF THE SYSTEM WHILE THE ACCURACY OF THE SYSTEM WAS CALCULATED USING ANOTHER FORMULA. THE RESULTS FROM BOTH THE MEASUREMENTS SHOWED A SATISFACTORY PERFORMANCE.

KEYWORDS: *H*UMAN-COMPUTER INTERACTION, HIDDEN MARKOV MODEL, ACOUSTIC MODEL, WORD ERROR RATE.

INTRODUCTION

A speech recognition system recognizes the words uttered by a speaker. Speech is one of the best methods to interact with computers rather than typing and clicking. It also acts as a better interface for illiterate as well as physically disabled people. Researchers and computer scientists are working towards making computers understand human speech. As a result of these, attempts are taken to develop a human and machine interface where both can communicate in an unskilful way.

Several researches have been carried out on speech recognition, but most of them have been done only for English, Chinese, Mandarin and Arabic languages. Very few researches have been carried out for Tamil language to recognize isolated speech. Tamil is the second major language spoken in Sri Lanka and therefore, recognition of Tamil speech is beneficiary. Recognition of Tamil language is a challenging task (Chong,et.al, 2008) due to the speech variability in any human's spoken utterance and language nature of Tamil Language (Kalith, et.al, 2016). This research is a first step for implementing an efficient speech recognition

[134] Postgraduate Institute of Science, University of Peradeniya, Sri Lanka.
[135] Postgraduate Institute of Science, University of Peradeniya, Sri Lanka.

system for Tamil speaking people in order to increase the human computer interaction in Tamil language.

Tamil is a Dravidian language spoken by 77 million people all over the world. It is a 15th largest language among the world languages. Spoken predominantly in south India Sri Lanka and Malaysia, it is an official language in India and Sri Lanka.

The research is mainly aimed to design and implement a Tamil speech recognition system based on Hidden Markov Model (HMM) using CMUSphinx toolkit. The system is developed for the commands which are used to execute computer applications.

The main activities involved are to feed the spoken utterance into the computer via an audio device as to train the system. Then the system is tested with spoken utterance to evaluate the recognition result. The recognition is a connected word of Tamil speech of about 125 computer commands.

The organization of the paper is given as follows. Some of the work related to speech recognition and methods used previously are discussed in Section 2 under the topic literary study. Section 3 deals with the Methodology of the proposed system. Speech recognition architecture, its design, Technology adopted, and approaches used are described in the 4th section. Section 5 describes about the Evaluation details. Finally, the paper concludes and proposes future work.

LITERARY STUDY

Current State of Speech Recognition Technology

Speech recognition technology is keeping on advancing its technology daily, taking many factors into consideration. Speech input may either be isolated speech, connected speech, spontaneous speech or continuous speech (Kalith, et.al, 2016) and the system may handle the speaker utterances accordingly.

Few systems have been developed and they allow only a limited form of natural language input within a very specific domain at any particular point in the interaction. ASR devices have been available from 1970s, but they were very expensive and could recognize only isolated speech. Later in 80s and 90s, ASR technology showed some improvements and reached to an advanced level in the late 90s where the researchers were able to develop software for desktop applications. They also developed two types of ASR: Speaker Dependent and Speaker Independent.

ASR also used various techniques for recognition of speech, and they are listed below.

Template based approaches matching
Words are recorded and saved in a repository. Unknown words uttered by a speaker are compared against these recorded words as to find the best matched word. Dynamic Type warping approach falls into this technique (Tolba & O'Shaughnessy, 2001).
Knowledge based approaches

This approach stores speech an expert knowledge on the variations of speech but this is not much effective.

Statistical based approaches (Hidden Markov Model-HMM)

The speech variations are modelled statistically and based on the probabilistic value; the words are matched.

Learning based approaches

This approach utilizes the neural network and genetic algorithm techniques. The models are studied through emulation or evolutionary process.

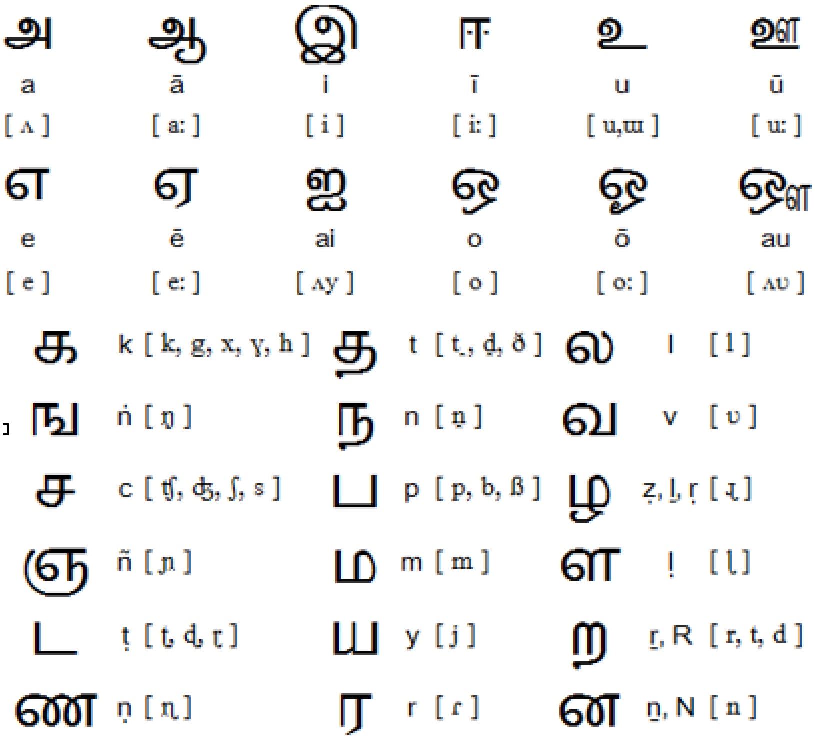
Artificial intelligence approach

This is an attempt to mechanize the way the words are recognized according to the user's perspective. This approach widely uses expert systems (Mori et.al., 1987).

Among all the ASR techniques mentioned above, we have adopted statistical based approach which is HMM. HMM is an effective method for developing a high-performance speaker independent speech recognition engine. This technique also is very suitable for large vocabulary (Ehsani & Knodt, 1998).

Tamil Speech Recognition System

Tamil language is spoken in many parts of the world and the majority is in the South Asia: India and Sri Lanka. A little number of researches has been carried out in the context of Tamil speech recognition. But they have only been done to recognize numbers and some isolated and continuous words or utterances. No systems have been developed for desktop applications for smart devices. Tamil language has 12 vowels and 18 consonants, but the number of phonemes counts more due to different sounds depending on where they occur (Gnanathesigar, 2012).



அ	ஆ	இ	ஈ	உ	ஊ
a	ā	i	ī	u	ū
[ʌ]	[a:]	[i]	[i:]	[u,ɯ]	[u:]
எ	ஏ	ஐ	ஒ	ஓ	ஔ
e	ē	ai	o	ō	au
[e]	[e:]	[ʌy]	[o]	[o:]	[ʌʋ]

க	k	[k, g, x, ɣ, h]	த	t	[t̪, d̪, ð]	ல	l	[l]
ங	ṅ	[ŋ]	ந	n	[n̪]	வ	v	[ʋ]
ச	c	[ʧ, ʤ, ʃ, s]	ப	p	[p, b, β]	ழ	ẓ, ḻ, ṛ	[ɻ]
ஞ	ñ	[ɲ]	ம	m	[m]	ள	ḷ	[ɭ]
ட	ṭ	[ʈ, ɖ, ɽ]	ய	y	[j]	ற	ṟ, R	[r, t, d]
ண	ṇ	[ɳ]	ர	r	[ɾ]	ன	ṉ, N	[n]

Fig. 1 Vowels and Consonants in Tamil Language

Implementation of ASR for Tamil language is not an easy task as speech recognition design depends on many other fields, namely acoustics, signal processing, pattern recognition, phonetics, linguistics, psychology, neuroscience, and computer science.

METHODOLOGY

Figure 1 below shows the process to design and implement the speech recognition system. The main aim of the research is to implement a speech recognition system for Tamil language for windows desktop applications. CMUSphinx toolkit was used for recognition process. CMUSphinx is an open source speech recognition toolkit known that comprises of libraries Sphinxbase, Pocketsphinx, and SphinxTrain.

Pocketsphinx - lightweight recognizer library written in C.
Sphinxbase - support library required by Pocketsphinx
SphinxTrain - acoustic model training tool

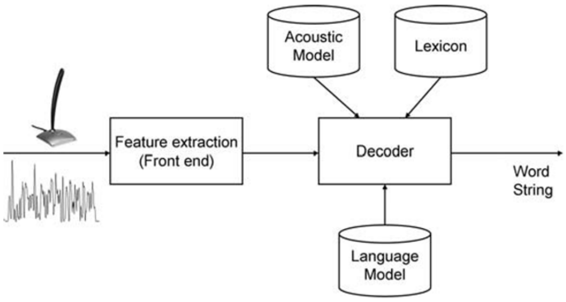


Fig. 2 Speech Recognition Process

Data Preparation

Voice recordings of commands used for executing PC applications were first collected from several people speaking Tamil language living in various parts of Sri Lanka were collected. During the collection of voice recordings, age, gender, and language dialect were mainly considered. Additionally, the environmental background and time were also taken into considerations. When all the recordings were taken, as a next step, the audio files were converted into wav files of mono channel at a sampling rate of 16kHz and a bi rate of 16. Later, the voice recordings were segmented into separate voice commands using Audacity. Figure 2 shows a segmented voice command.

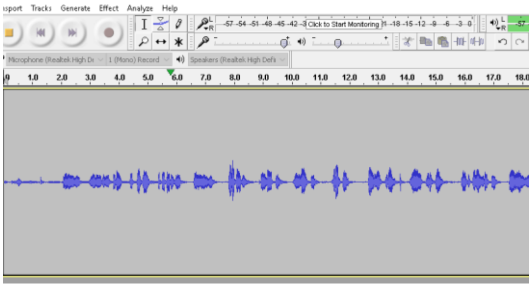

sport Tracks Generate Effect Analyze Help
Click to Start Monitoring
Microphone (Realtek High De
1 (Mono) Record
Speakers (Realtek High Defi
1.0 2.0 3.0 4.0 5.0 6.0 7.0 8.0 9.0 10.0 11.0 12.0 13.0 14.0 15.0 16.0 17.0 18.0

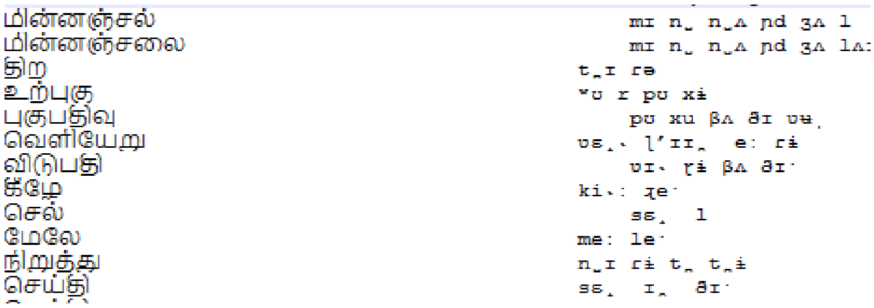
Fig. 3 Segmentation of Voice Recording using Audacity

Language Model

Language model represents the most likelihood occurrence of a word uttered. It works based on a probability and used to search for a word and improve accuracy. It lists 1-,2- and 3-grams along with their likelihood (the first field) and a back-off factor (the third field). SRILM toolkit was used to build the language model for Tamil language.

Lexicon Model

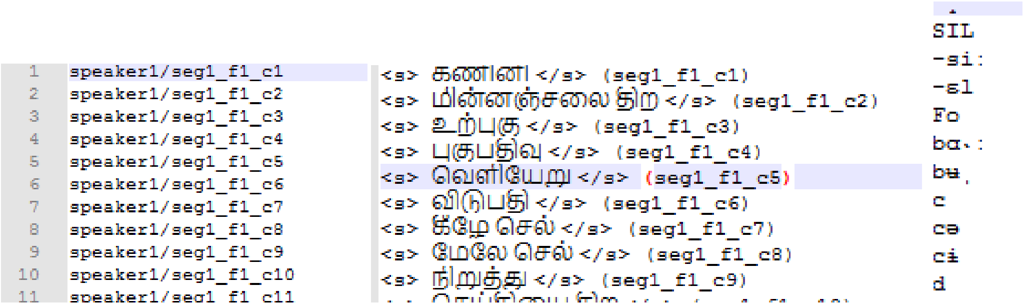
Lexicon Model also known as Pronunciation dictionary, describes how each word are pronounced. Each and every word the recognizer needs to recognize must be placed in the dictionary file. That is a dictionary file contains the words to be recognized and their respective monophonic sequences. Here, pronunciation dictionary for Tamil language has been constructed manually for each word in the vocabulary. 'Anunaadam' tool, an online IPA converter was used to create the monophonic sequence of the words.



```
மின்னஞ்சல்                          mɪ n̪ n̪ʌ ɲd ʒʌ l
மின்னஞ்சலை                          mɪ n̪ n̪ʌ ɲd ʒʌ lʌː
திற                              t̪ɪ rə
உற்புகு                           ʷʊ r pʊ xɨ
புகுபதிவு                           pʊ xu βʌ ðɪ ʋʉ
வெளியேறு                          ʋɛ ɭʼɪɪ eː rɨ
விடுபதி                            ʋɪ ɽɨ βʌ ðɪ
கீழே                              kiː ɻe
செல்                                sɛ l
மேலே                             meː le
நிறுத்து                            n̪ɪ rɨ t̪ t̪ɨ
செய்தி                             sɛ ɪ ðɪ
```

Acoustic Model

An acoustic model was trained for Tamil language for the training as well as testing purpose. In order to train acoustic model, fields file, transcription file, dictionary file, and file containing phone sounds (phone file) were prepared. Sphinxtrain tool was used to obtain the acoustic model.

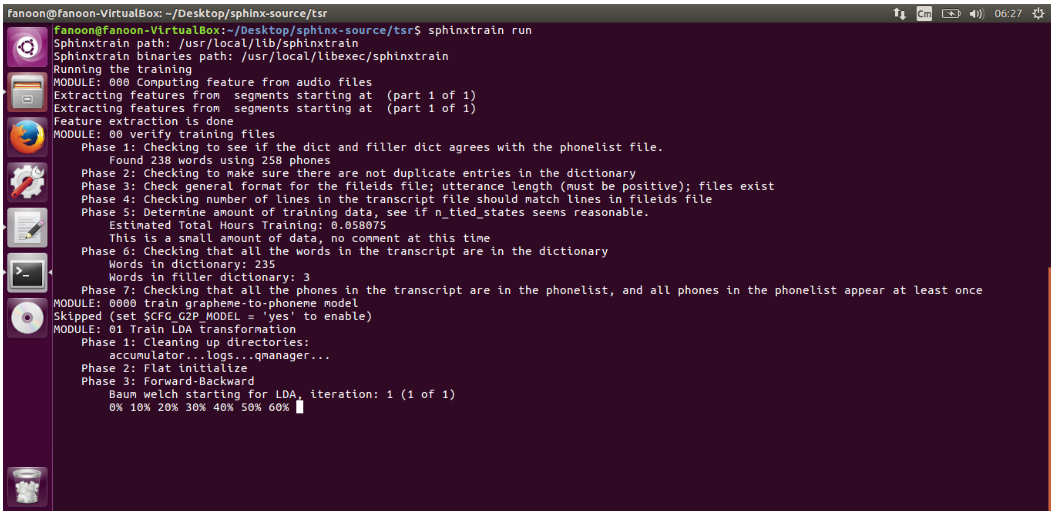


```
speaker1/seg1_f1_c1
speaker1/seg1_f1_c2
speaker1/seg1_f1_c3
speaker1/seg1_f1_c4
speaker1/seg1_f1_c5
speaker1/seg1_f1_c6
speaker1/seg1_f1_c7
speaker1/seg1_f1_c8
speaker1/seg1_f1_c9
speaker1/seg1_f1_c10
speaker1/seg1_f1_c11
```

```
<s> கணானா </s> (seg1_f1_c1)
<s> மின்னஞ்சலை திற </s> (seg1_f1_c2)
<s> உற்புகு </s> (seg1_f1_c3)
<s> புகுபதிவு </s> (seg1_f1_c4)
<s> வெளியேறு </s> (seg1_f1_c5)
<s> விடுபதி </s> (seg1_f1_c6)
<s> கீழே செல் </s> (seg1_f1_c7)
<s> மேலே செல் </s> (seg1_f1_c8)
<s> நிறுத்து </s> (seg1_f1_c9)
```

```
SIL
-siː
-ɛl
Fo
bɑː
bʉ
c
cə
cɨ
d
```

Fig. 5 fileid file, transcription file and phone file



```
fanoon@fanoon-VirtualBox:~/Desktop/sphinx-source/tsr$ sphinxtrain run
Sphinxtrain path: /usr/local/lib/sphinxtrain
Sphinxtrain binaries path: /usr/local/libexec/sphinxtrain
Running the training
MODULE: 000 Computing feature from audio files
Extracting features from  segments starting at  (part 1 of 1)
Extracting features from  segments starting at  (part 1 of 1)
Feature extraction is done
MODULE: 00 verify training files
    Phase 1: Checking to see if the dict and filler dict agrees with the phonelist file.
        Found 238 words using 258 phones
    Phase 2: Checking to make sure there are not duplicate entries in the dictionary
    Phase 3: Check general format for the fileids file; utterance length (must be positive); files exist
    Phase 4: Checking number of lines in the transcript file should match lines in fileids file
    Phase 5: Determine amount of training data, see if n_tied_states seems reasonable.
        Estimated Total Hours Training: 0.058075
        This is a small amount of data, no comment at this time
    Phase 6: Checking that all the words in the transcript are in the dictionary
        Words in dictionary: 235
        Words in filler dictionary: 3
    Phase 7: Checking that all the phones in the transcript are in the phonelist, and all phones in the phonelist appear at least once
MODULE: 0000 train grapheme-to-phoneme model
Skipped (set $CFG_G2P_MODEL = 'yes' to enable)
MODULE: 01 Train LDA transformation
    Phase 1: Cleaning up directories:
        accumulator...logs...qmanager...
    Phase 2: Flat initialize
    Phase 3: Forward-Backward
        Baum welch starting for LDA, iteration: 1 (1 of 1)
        0% 10% 20% 30% 40% 50% 60%
```

Fig. 6 Acoustic Model Training in ubuntu

Speech Recognition on Windows

All the above processes were carried on in Ubuntu environment and then, the output was implied onto the windows environment. For this purpose, CMUSphinx was installed onto windows with other dependencies such as MS Visual Studio, Swig, Python and many other. After the successful setting up of the environment on windows, two additional files argFile.txt and ctlFile.txt were created inside the Pocketsphinx.

argFile.txt consisted of the path to the acoustic model, language model and the lexicon model.

ctlFile.txt consists of all the utterances for the speech recognition; i.e., it has all the commands recorded.

And finally, the models were integrated with another Speech Recognition System Simon that could control the Windows applications.

RESULTS AND DISCUSSION

Hidden Markov Model was used for recognition purpose and both known and unknown speakers were selected. They were asked to utter 15 commands from the recorded list and the number of commands recognized was identified. The result is tabulated below.

Table 1 Results of Speech Recognition

Speakers	**Number of commands**	**Correct recognition**	**Incorrect recognition**
Known speaker	15	12	3
Unknown speaker	15	8	7

The Word Error Rate (WER) of the system was calculated with the formulae;

WER = (I+D+S)/N

Where I, D, and S are the number of words inserted, deleted and substituted respectively. This system had nearly three hundred words in total and only fifty of them were used for testing.

Table 2

The number of speakers against the accuracy

Speaker	Accuracy	Speaker	Accuracy
First	60%	Sixth	78%
Second	70%	Seventh	80%
Third	65%	Eighth	85%
Fourth	73%	Ninth	90%
Fifth	75%	tenth	75%

And the accuracy of the Tamil speech recognition system was calculated with the formula;

Accuracy = (N-D-S)/N

According to table, Table 02, it is very clear that accuracy depends on the training time and also may vary depending on the speaker and the environment where the words are uttered.

CONCLUSION

Speech recognition system was built for a very few commands and tested with each 5 male and female speakers. The database size used for this research work is approximately 250 words and produces an average of 85% of accuracy. Our vocabulary size is small and therefore, the system gives lesser word error rate compared to the system with large size vocabulary. Speech recognition is generally easy when vocabularies are small, but word error rate increases as the vocabulary size grows.

In future, this project can be used at a very large scale and be integrated with desktop applications. The speaker amount can be increased up to 200 including both male and female speakers with 5 hours of recording in order to get a speech recognizer with maximum accuracy. Also, the research can be further extended to increase the accuracy by parallelizing the speech recognition tasks, i.e. the search algorithm, using GPGPU processors.

REFERENCES

Chong, J. et al., 2008. *Data-Parallel Large Vocabulary Continuous Speech,* Berkeley: s.n

CMUSphinx Tutorial For Developers'**,** wiki article, [Online]. Available: http://cmusphinx.sourceforge.net/wiki/tutorial, [Accessed: 24th of May 2016]

Ehsani, F, & Knodt, E. 1998, 'Speech Technology In Computer Aided Learning: Strengths And Limitations Of A New Call Paradigm', Langugae Learning and Technology, [Online]. Available: http://hstrik.ruhosting.nl/wordpress/wp-content/uploads/2013/03/Ehsani-Knodt-LLT-1998.pdf

Gnanathesigar.H, 2012, 'Tamil Speech Recognition using Semi-Continuous Models', *International Journal of Scientific and Research Publications,* vol. 2[6]

Installing CMU-Sphinx on Ubuntu, [Online]. Available:

http://jrmeyer.github.io/installation/2016/01/09/Installing-CMU-SPhinx-on-Ubuntu.html Josh Meyer's Website, [Accessed: 15th of August 2016]

Kalith, M., David, A. & Samantha, T. (2016), 'Isolated to Connected Tamil Digit Speech Recognition System Based on Hidden Markov Model', *International Journal of New Technologies in Science and Engineering*, 3(4), 2-6

Lee, K. F., n.d. In: *Automatic Speech Recognition.* Boston/London/Dodrecht: Kluwer Academic Publishers

Mori R.D, Lam L., and Gilloux M. (1987). Learning and plan refinement in a knowledgebased system for automatic speech recognition. IEEE Transaction on Pattern Analysis Machine Intelligence, 9(2):289-305

Picheny, M., (2002). Large vocabulary speech recognition, IEEE Computer, 35(4),42- 50

Reddy D.R., (1976). Speech Recognition by Machine: a Review. Proceeding of IEEE, 64(4),501-531

Shammur, A.C 2010, 'Implementation of Speech Recognition System for Bangla: thesis, BRAC University

SRILM Installation and Running Tutorial, [Online]. Available: http://okapiframework.org/wiki/index.php?title=SRILM_Installation_and_Running_Tutorial [Accessed: 20th of August 2016]

Tamil, Omniglot, the online encyclopedia for writing systems and languages, [Online]. Available: http://www.omniglot.com/writing/tamil.htm , [Accessed: 15th of November 2016]

Tolba, H., and O'Shaughnessy, D., (2001)., Speech Recognition by Intelligent Machines, IEEE Canadian Review (38).

Vrinda, & Chander,S, T 2003, 'SPEECH RECOGNITION SYSTEM FOR ENGLISH LANGUAGE', *International Journal of Advanced Research in Computer and Communication Engineering,* vol. 2, pp. 919-922.

Waibel, A. & Kai-Fu Lee, n. d. In: M. B. Morgan, ed. *Readings in Speech Recognition*. San Mateo, California: Morgan Kaufmann Publishers Inc...

Y. Anunaadam, [Online]. Available from:

http://anunaadam.appspot.com/, [Accessed: 25th of May 2016].