International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 10, October- 2014

Sanskrit Speech Recognition using Hidden Markov Model Toolkit

Jitendra Singh Pokhariya
Electronics&communication
G.B.P.U.AT Pantnagar Uttarakhand
India.

Dr. Sanjay Mathur
Electronics&communication
G.B.P.U.AT Pantnagar Uttarakhand
India

Abstract: **Automated Speech Recognition (ASR) is the ability of a machine or program to recognize the voice commands or take dictation which involves the ability to match a voice pattern against a provided or acquired vocabulary. At present, mainly Hidden Markov Model (HMMs) based speech recognizers are used. This research work aims to build a speech recognition system for Sanskrit language. Hidden Markov Model Toolkit (HTK) is used to develop the system. The system is trained to recognize 50 Sanskrit utterances. Training data has been collected from ten speakers. The experimental results show that the overall accuracy of the presented system with 5 state and 10 states in HMM topology is 95.2% to 97.2% respectively.**

Keywords: HMM; HTK; Mel Frequency Cepstral Coefficient (MFCC); Automatic Speech Recognition (ASR); Sanskrit;

I. INTRODUCTION

There are number of ways through which human can communicate with machine, for example to communicate with computer a keyboard is needed. But for handicapped people it has always been a problem to communicate with machine as computer etc. For example, blind people cannot type with ordinary keyboard. Speech interface can help to handle these problems. Now a days, due to its versatile application, speech recognition is a promising field of research. Speech synthesis and Speech recognition together form a speech interface. A speech synthesizer converts text into speech. Thus it can read out textual contents from the screen. Speech recognizer has ability to understand the spoken word and convert it into text. This research work is based on speech recognition. Speech recognition refers to the ability to listen spoken words and identify various sounds present in it, and recognize them as the words of a known language. There are various steps required to perform speech recognition and they are: voice recording, word boundary detection, feature extraction, and recognition with the help of knowledge model. Word boundary detection is the method of identifying the start and the end of a spoken word in a given speech signal. While analyzing the speech signal, at times it become difficult to identify the word boundary. This depends on the various factors like, accents, time duration of the pause given between the words while speaking.

Feature extraction refers to conversion of speech signal in its raw form to other form, representing characteristic information contained in it e. g. speech of "WAV" format into "MFCC" format for further use in the process of training and testing. Here MFCC represents Mel Frequency Cepstral Coefficient. Feature extraction also includes extraction of parameters such as amplitude and energy of signal. Recognition involves mapping of given test signal to one of the training data. This involves use of various knowledge model(10). Knowledge model(2) refers to a model such as phone acoustic model, language model. Phone model can be monophone and triphone(13). In this research work a language model() is developed for Sanskrit language. In order to generate the language model the system needs to be trained. After 40 year of research, speech recognition is still a state of art. It has a problem of noise(12) as well as grammar and accent, which differs with different language. Sanskrit language is used for this research work because it has 444 phonems and it is also a highly inflectional (alteration in pitch) language. Each word of this language is inflected before using in a sentence, work can be a noun or a verb Inflection means each word produces different pitch due to this the problem homonyms is rectified in this language. For training and recognition purpose Hidden Markov Model Toolkit (HTK) is used which is a portable toolkit for building and manipulating Hidden Markov Model (HMM). HTK is primarily used for speech recognition although it has been used for numerous other application including research into speech synthesis, character recognition and DNA sequencing. HTK was originally developed at the Machine Intelligence Laboratory of the Cambridge University Engineering Department(CUED)in 1989, where it has been used to build CUED's large vocabulary speech recognition system.

1.1 Motivation

The reasons ranging from technological curiosity about the mechanisms for mechanical realization of human speech capabilities, to the desire to automate simple tasks inherently requiring human-machine interactions, research in automatic speech recognition (and speech synthesis) by machine has attracted a great deal of attention over the past five decades. Many international organizations like Microsoft, SAPI and Dragon-Naturally-Speech as well as research groups are working on this field especially for European languages.

IJERT

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

II. RELATED WORKS

This section presents some of the reported works available in the literature that are similar to the presented work. Dua et. al. (2012) presented the implementation of an isolated word Automatic Speech Recognition system (ASR) for an Indian regional language Punjabi. The HTK toolkit based on Hidden Markov Model (HMM), a statistical approach, is used to develop the system. Initially the system is trained for 115 distinct. Punjabi words by collecting data from eight speakers and then is tested by using samples from six speakers in real time environments. Saini et. al. (2013) proposed a speech recognition system for Hindi language. Hidden Markov Model Toolkit (HTK) is used to develop the system. It recognizes the isolated words using acoustic word model. The system is trained for 113 Hindi words. Training data has been collected from nine speakers.

III. MODEL

Sanskrit is a highly inflectional language. Every word, be it a noun or a verb has to be inflected before it is used in a sentence. Before explaining more about Sanskrit language, first we have to explain more about ASR. Why is automatic speech recognition such a difficult problem that, after 40 years of research, it still has not been solved? At first sight it may seem just a matter of classifying sounds using some typical characteristics of these sounds. This approach, called acoustic approach(1) was indeed tried, but only with limited results. A second approach to recognizing, that does not directly rely on a set of characteristics is the statistical pattern recognition approach, which has already been successfully been applied to problems like the automatic recognition of handwriting. The pattern recognition approach did prove to be fruitful, but only after advanced models were developed. Figure 1 schematically shows the speech recognition problem: someone produces some speech and we want to have a system (the box in the figure) that automatically translates this speech, a pressure waveform that is, to a written transcription.

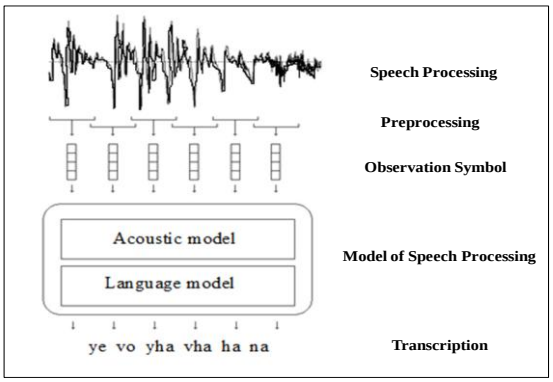

Fig.1 Speech recognition model

IV. HIDDEN MARKOV MODEL(10)

Markov Model is a process in which each state corresponds to a deterministically observable event, and hence the output of any given state is not random. Hence Markov Models are too restrictive to be applicable to many practical problems including speech recognition. The concept of Markov Models to include the case in which the observation is a probabilistic function of the state. That is the resulting model is a doubly embedded stochastic process(7) with an underlying stochastic process that is not directly observed only through another set of stochastic process that produce the sequence of observation.

It is possible, after some preprocessing, to represent the speech signal as a sequence of observation symbols $O=o_1 o_2 \ldots o_T$ that represents a string composed of elements of a alphabet V of symbols. If, in addition, we have a vocabulary V of all the words w_i, $1 \le i \le |V|$, that can be uttered. Then mathematically the speech recognition problem comes down to finding the word. Sequence $\hat{W}$ having the highest probability of being spoken, given the acoustic evidence O, thus we want to solve

$$\hat{W} = arg\max_W P(W|O) \quad (1)$$

Unfortunately, this equation is not directly computable since the number of possible observation sequences is sheer inexhaustible, unless there is some limit on the duration of the utterances and there is a limited number of observation symbols. But Bayes formula gives(12):

$$P(W) = \frac{P(W).P(O|W)}{P(O)} \quad (2)$$

Where $P(W)$, called the language model, is the probability that the word string W will be uttered and $P(O|W)$is the probability that when word string W is uttered the acoustic evidence O will be observed, the latter is called the acoustic model. The probability $P(O)$ is usually not known but for a given utterance it is of course just a normalizing constant and can be ignored. Thus to find a solution to formula (1) we have to find a solution to():

$$\hat{W} = arg\max_W P(O|W) \quad (3)$$

Consequently, a speech recognizer consists of three components: a preprocessing part that translates the speech signal into a sequence of observation symbols, a language model that tells us how likely a certain word string is to occur and an acoustic model that tells us how a word string is likely to be pronounced. In the next sections these three subsystems will be described.

V. HTK

HTK is a portable software toolkit for building and manipulating systems that use continuous density Hidden Markov models. It has been developed by the Speech

IJERT

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

Group at Cambridge University Engineering Department. HMMs can be used to model any time series and the core of HTK is similarly general purpose. However, HTK is primarily designed for building HMM based speech processing tools, in particular speech recognizers. In can be used to perform a wide range of tasks in this domain including isolated or connected speech recognition using models based on whole word or sub-word units, but it is especially suitable for performing large vocabulary continuous speech recognition.HTK includes nineteen tools that perform tasks like manipulation of transcriptions, coding data, various styles of HMM training including Baum-Welch re-estimation, Viterbi decoding, results analysis and extensive editing of HMM definitions.

V. SANSKRIT CHARACTER SET

The word Sanskrit means "most perfect" because not a single letter, word, or verse can be pronounced without having bona fide principle. The first principle, which is hardly seen in any other language, is that for every sound there is only one sound. There are five places for pronunciation: the throat, the palate, the upper part of the palate, the teeth, and the lips. The pronunciation types are shown in table 1. In pronouncing vowels a sound is produced by air by these different organs and the tongue. The air is not stopped or blocked at any point. With consonants the sound is produced in a similar way, but the tongue or the lips make a full contact, stopping and releasing the air.

1. Kanthya	a	ă	ka	kha	ga	gha
2.Talavya	i	i̯	ca	cha	ja	jha
3.murdhanya	r̥	r̲	ṭa	ṭha	ḍa	ḍha
4. Dantya	l̥	l̲	ta	tha	da	dha
5.Osthya	u	ŭ	pa	pha	ba	bha

Table 1 Five pronunciation types of speech

Sanskrit grammar has distinguished the terms *varna* (phoneme) and *akshara* (syllable). Both these terms are used in the context of spoken languages and can be ex-tended to written languages. Since the oral tradition in India was of a higher order, the stress on right pronunciation was laid at most on the spoken language. To represent such speech nuances in written language, various *chinhas* (signs) were
evolved as to strike the equivalence in spoken and written ex-pressions. This extraordinary activity is part of the Indian tradition.Therefore, the realization of such phonemic system in the context of new technology seems to be imperative where writing is talked in the context of speech and speech in the context of writing. The attempt is made to identify *varnamala* comprising of basic speech sound units as vowel phonemes (*swaravarna*) and consonant phonemes (*vyanjanvarna*). Presently, Devanagari script is used for writing classical Sanskrit as well as Vedic Sanskrit. This includes the multitier usage of diacritic marks of complex compositions, above, below and at the sides of the base glyphs. Therefore, as compared to modern historical derivatives from Sanskrit such as Hindi, Marathi, Nepali etc., the Sanskrit text demands adequate range of characters as well as exhaustive rendering rules to achieve the advanced typographic quality in Classical as well as Vedic Sanskrit text. As mentioned earlier phonemes are divided into two types: vowel phonemes (*swaravarna*) and consonant phonemes (*vyanjanvarna*). They together broadly constitute the *Varnamala* which has been referred as a *varna-samamnaya*. The orthographic representation of these varnas is done in a systematic way. The combination of consonant phoneme and a vowel phoneme produces a syllable (*akshara*). A cluster of glyphs emerges as an outcome of this process.

VI. IMPLEMENTATION

The steps of implementation of speech recognition is described here.Hindi Speech recognition system is developed using HTK toolkit on the Linux platform. HTK v3.4 and ubuntu13.04 are used. Firstly, the HTK training tools are used to estimate the parameters of a set of HMMs.

6.1 Data Interpretation

Training and testing a speech recognition system needs a collection of utterances. In this work total of 10 distinct speakers are used for this thesis work and each one is asked to spoke 50 utterances in Sanskrit. The 50 utterances are divided into 15 grammar models for training and testing purpose. The 50 utterances are consist total 198 words, and character length of words from 2 to 6 characters. Out of 10 speakers 6 speakers are used in both training and testing, and 4 speakers are used only in testing. The utterance accuracy for each speaker is the average of utterance accuracies of the 15 grammar models.

6.2 Task Grammar

For a single speaker fifteen grammar models are created, and the training and testing for each grammar model is done separately. As an example a grammar model named as "gram" shown below. A "gram" file is used to create word network using HParse tool.

```
$word = kalyaana roopaaya kalow janaanaam
| naraayne thyaadhi japadhbi rucchai |
bhakthai sadhaa poorna mahaalayaaya;
(SENT-START <$word> SENT-END)
```

6.3 Feature Extraction(4)

In order to generate MFFC feature file(.mfc) a configuration file(config) is formed. The configuration file contains coding parameters. The Configuration file is shown in table 2. The coding parameters are used by tool HCopy. Speech feature like "MFCC" and Spectrogram are very important in explanation of any language used for recognition. In this thesis for modeling purpose 39 speech

IJERT

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

feature are used. Out of which 12 are MFCC, a "C0" (energy component of whole sample) and 13 acceleration, 13 delta coefficient. These are time derivatives parameter. The performance of speech recognition is greatly enhanced by adding time derivative parameter. In HTK these are indicated by attaching qualifiers to basic parameters. The qualifier "D" stand for first order regression coefficient and "A" stand for second order regression coefficient.

SOURCEKIND	WAVEFORMAT
TARGETKIND	MFCC_0_D_A
TARGETRATE	100000.0
SAVECOMPRESSED	T
SAVEWITHCRC	T
WINDOWSIZE	250000.0
USEHAMMING	T
PREEMCOEF	0.97
NUMCHANS	26
CEPLIFTER	22
NUMCEPS	12

Table 2 Configuration parameters

6.4 Training the HMM

Before starting the training process, the HMM parameters must be properly initialized with training data in order to allow a fast and precise convergence of the training algorithm. HTK offers a initialization tool: HCompv. The HCompv tool performs a "flat" initialization of a model. Every state of the HMM is given the same mean and variance vectors: these are computed globally on the whole training corpus. After Initialization the tool HERest is used to perform a single re-estimation of the parameters on a set of HMMs using an *embedded training* version of the Baum-Welch algorithm. For each training utterance, a composite model is effectively synthesized by concatenating the phoneme models given by the transcription.

6.5 RESULTS

The performance of the system is tested against speaker independent parameter by using two types of speakers: one who are involved in training and testing both and the other who are involved in only testing. The second parameter for checking system performance by varying no. of states in HMM topology.The table 3 to 8 shows the evaluation results of the Sanskrit speech recognition system. A total of 10 distinct speakers are used for this thesis work and each one is asked to spoke 50 utterances in Sanskrit. The 50 utterances are divided into 15 grammar models for training and testing purpose. The 50 utterances are consist total 198 words, and character length of words from 2 to 6 characters. Out of 10 speakers 6 speakers are used in both training and testing, and 4 speakers are used only in testing. The experimental results show that the overall accuracy of the presented system with 5 state and 10 states in HMM topology is 95.2% and 97.2% respectively.

VII. CONCLUSION

The objective of this study is to build a speech recognition system for Sanskrit language using HTK toolkit on the Linux platform. The work may further be extended to large vocabulary size and to spontaneous speech recognition.

IJERT

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

Recognition by speaker involved in both training and testing:

Speaker Number	No. of spoken utterances	Length of utterance (in word)	No. of states in HMM topology	No. of Recognized utterances	% utterance accuracy	utterance error rate
S1	50	3-6	5	48	96	4
S2	50	3-6	5	50	96	4
S3	50	3-6	5	47	94	6

Table 3 5 state HMM Results for speaker s1,s2,s3.

Recognition by speaker involved in both training and testing:

Speaker Number	No. of spoken utterances	Length of utterance (in word)	No. of states in HMM topology	No. of Recognized utterances	% utterance accuracy	utterance error rate
S1	50	3-6	10	50	100.	0
S2	50	3-6	10	50	100	0
S3	50	3-6	10	48	96	4

Table 4 10 state HMM Results for speaker s1,s2,s3

Recognition by speaker involved in both training and testing:

Speaker Number	No. of spoken utterances	Length of utterance (in word)	No. of states in HMM topology	No. of Recognized utterances	% utterance accuracy	utterance error rate
S4	50	3-6	5	47	94	6
S5	50	3-6	5	50	100	0
S6	50	3-6	5	47	94	6

Table 5 5 state HMM Results for speaker s4, s5 ,s6.

IJERT

Recognition by speaker involved in both training and testing:

Speaker Number	No. of spoken utterances	Length of utterance (in word)	No. of states in HMM topology	No. of Recognized utterances	% utterance accuracy	utterance error rate
S4	50	3-6	10	50	100	0
S5	50	3-6	10	50	100	0
S6	50	3-6	10	48	96	0

Table 6 10 state HMM Results for speaker s3, s5, s6

Recognition by speaker involved only in testing:

Speaker Number	No. of spoken utterances	Length of utterance (in word)	No. of states in HMM topology	No. of Recognized utterances	% utterance accuracy	utterance error rate
S7	50	3-6	5	46	92	8
S8	50	3-6	5	47	94	6
S9	50	3-6	5	47	94	6
S10	50	3-6	5	47	94	6

Table 7 5 state HMM Results for speaker s7, s8, s9, s10.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

Recognition by speaker involved only in testing:

Speaker Number	No. of spoken utterances	Length of utterance (in word)	No. of states in HMM topology	No. of Recognized utterances	% utterance accuracy	utterance error rate
S7	50	3-6	10	47	94	6
S8	50	3-6	10	47	94	6
S9	50	3-6	10	48	96	4
S10	50	3-6	10	48	96	4

Table 8 10 state HMM Results for speaker s7, s8, s9, s10.

REFERENCES

1. **Aggarwal, R.K., and Dave, M.,** "Acoustic modeling Problem for Automatic Speech Recognition System: Conventional Methods International Journal of Speech Technology 2011 vol.14 pp. 297–308.
2. **Das, P. K., Tripathy, H. K., Tripathy, B. K.,** "A Knowledge based Approach Using Fuzzy Inference Rules for Vowel Recognition", Journal of Convergence Information Technology vol. 3 No 1, March 2008.
3. **Hemdal, J.F., and Hughes, G.W.,** "A feature based computer recognition program for the modeling of vowel perception, in Models for the Perception of Speech and Visual Form, pp. 440-453 MIT Press, Cambridge, MA, 1964".
4. **Jain, R. And Saxena, S. K.,** "Advanced Feature Extraction & Its Implementation In Speech Recognition System" IJSTM, vol. 2 Issue 3, July 2011.
5. **Khadyan, V., Aggarwal, R. K.,** "Punjabi automatic speech recognition system" " International Journal of computer science vol.9 July 2012.
6. **Lee, K.S.,** "EMG-Based Speech Recognition Using Hidden Markov Models With Global Control Variables" IEEE Transactions on Biomedical Engineering, vol. 55, issue-3, pp: 930-940, March 2008.
7. **Ostendorf, M., Digalakis, V., Kimball, O. A.,** "From HMM"s to segment models: a unified view of stochastic modeling for speech recognition". IEEE Transactions on Speech and Audio Processing, 1996 vol.4 pp. 360– 378 .
8. **Rabiner, L., Juang, B. H., Yegnanarayana, B.,** "Fundamentals of Speech Recognition", Pearson Publishers, 2010.
9. **Rabiner, L.,** "A tutorial on hidden Markov models and selected application in speech recognition,"Proc. IEEE, vol. 77, no. 2, pp. 257–286, Feb. 1989.
10. **Samoulian, A.,** "Knowledge Based Approach to Speech Recognition", IEEE Int. Conf. Acoust., Speech, Signal Processing, Adelaide Australia, April 1994, pp 177-180.
11. **Saini, P., Kaur, P., Dua, M.,** " Hindi automatic speech recognition system" International Journal of Engineering Trends and Technology (IJETT) – vol.4 Issue6 June 2013.
12. **Sankar A., Lee C. H., 1996** "A maximum-likelihood approach to stochastic matching for robust speech recognition"Speech and Audio Processing, IEEE Transactions on vol. 4 may 1996 pp 190-202".
13. **Thangarajan, R., Natarajan, A.M., Selvam, M.,** "Word and Triphone Based Approaches in Continuous Speech Recognition for Tamil Language",WSEAS Transactions on Signal Processing vol. 4 Issue 3, March 2008 pp 76-85.
14. **Tripathy, H. K., Tripathy, B. K., Das, P. K.,** "A Knowledge based Approach Using Fuzzy Inference Rules for Vowel Recognition", Journal of Convergence Information Technology vol. 3 No 1, March 2008
15. **Tripathy, S., Nandi, G.C., "**A MFCC based Hindi speech recognition technique using HTK Toolkit" International conference on Image Information Processing (ICIIP), IEEE Second International Conference 2013 pp. 539-544

IJERT

(This work is licensed under a Creative Commons Attribution 4.0 International License.)