International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 2, February - 2014

A Hindi Speech Recognition for Android OS

Saloni Sharma
M.Tech (Information Technology), Banasthali University
Jaipur, India

***Abstract*-- In the recent years, Mobile phone has become a staple and necessity in Indian society and around the globe. Its usage is spreading rapidly and has now become an everyday item. Mobile phone today, can serve a wide variety of needs besides being just a phone. Mobile phone can be used to make calls, store contact information, make to-do lists and reminders, and have a simple calculator. These are only the simplest of amenities. Today, technology scene is growing very fast and Smart Phones are spreading faster than any technology in human history. So today everyone wants to use the improved version of existing one.**

This project is based on the conversion of speech into text, in other words messages can voice/speech typed. In this paper there is voice to text conversion, first we will record sound through microphone and then word detection can be done by using "Pratt" software and then feature extraction can be done with the help of "Pratt" tool.

Keywords: Hidden Markov Model (HMM), Pratt tool.

I. INTRODUCTION

The main aim of the proposed work will be to develop an application for the Android phone which translates spoken word into text. The idea is taken from the project done by Ripul Gupta [5]. We have already studied the processing how we can create it and we will soon implement it to my project for Android System. As we know Android Smartphone user are increasing day by day and its demands is very high. So, today every customer wants to use Android Smartphone and programmers also wants to work on this platform because of its demand.

In this project we will create speech recognition software that can recognize Hindi words. This software development includes the study of Hidden Markov model (HMM). In this we will use "Pratt" software for taking sound signal parameter and removing noise. In this we talk about modules Sound recorder, Feature extractor and HMM training.

Android:

Today, everyday users are using Smartphone because of its unique features. Android operating system based on the Linux kernel (Monolithic-modifies Linux kernel), designed mainly for touch screen mobile devices like smart phone and tablet is attracting more and more customers and programmers too. Android is an open-source programmed in C (core), C++, java (UI) and its source code is released by Google under the Apache License. A survey conducted in April-May 2013, found that Android is the most popular platform used by 71% of the mobile developer. Since now November 2013, Android's share of the global Smartphone market, led by Samsung products, has reached 81%. As we can see that Android is the world's most widely used Smartphone. There are many languages which are used for speech recognition like CMU sphinx toolkit but these software's are only use to recognize English language or many other non Indian languages but not any Indian language. This project will help handicapped person who can write text by speak. He will not require typing anything, he will speak and it will write automatically.

II. EXISTING SYSTEMS

Although some promising features are available for Speech synthesis and Speech recognition, most of them are tuned to English. The acoustic and language model for these systems are for English language. Most of them require a lot of configuration before they can be used. ISIP and Sphinx are two of the known speech recognition software in open source.

III. SPEECH RECOGNITION

Smart phone offer their customers enhanced methods to interact with their cell phone but still there is something which makes their phone more intractable remains speech. Speech recognition is the translation of spoken words in text; we can also call it as "Automatic speech recognition", "Computer search recognition", "Speech to text" etc. Speech recognition refers to the ability to accept spoken words in audio format (wav or raw) and generate its content in text format.

Vocalizations vary in terms of pronunciation, accent, volume, pitch, articulation, roughness, speed and nasality. Speech is distorted by echoes and background noises. Accuracy of speech recognition varies with:

- Vocabulary size and confusability.
- Isolated, discontinuous and continuous speech.
- Adverse conditions
- Task and language constraints.

The steps involve making computer perform speech recognition are: Voice recording and word detection, feature extraction and recognition with the help of knowledge models. These are few components of Recognition system:

Voice Recording & Word detection:

The component is responsible from talking input from microphone and identifying the presence of word. Word detection is done by using energy and zero crossing rate of the signal. The output of this component is wave file or a direct feed from the feature extractor.

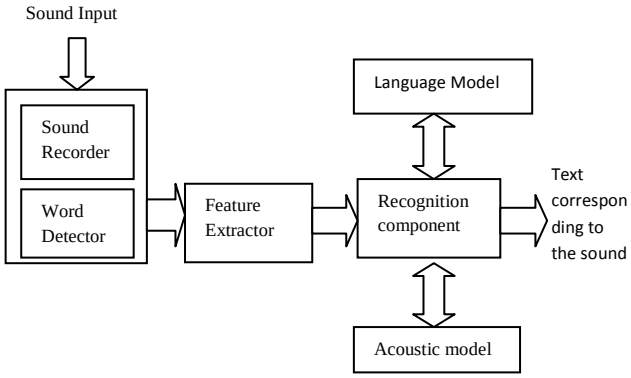

Fig. 2 Block Diagram of Training System

Features extraction:

It refers to the process of conversion of sound signal may include parameter such as amplitude of the signal, energy of frequency etc. This component generate feature vector for the sound signals given to it. It generates Mal frequency Cestrum Coefficient and Normalized energy as the feature that should be used to uniquely identify the given sound signal.

Recognition:

It involves mapping the given input to one of the known sounds. This may involve the use of various knowledge models for precise identification and ambiguity removal. It is a Continuous, Multi-dimensional Hidden Markov Model based component. It is the most important component of the system and is responsible for finding the best match in the knowledge base, for the incoming feature vectors.

Knowledge model:

It refers to models such as phone acoustic model, language models etc. which help the recognition system. To generate the knowledge model one needs to train the system. During the training period one needs to show the system a set of inputs and what output they should map to, this is often called as supervised learning. This component consists of word based Acoustic. Acoustic Model has a representation of how a word sounds. Recognition system makes use of this model while recognizing the sound signal.

Once the training is done can be summarized as the sound input which is taken from the sound recorder and is feed to the feature extraction module. The feature extraction module generates feature vectors out of it which are then forwarded to the recognition components. The recognition component with the help of the knowledge model and comes up with the result.

During the training the flow differs after generation of feature vector. Here the system takes the output of the feature extraction module and feeds it to the recognition system for modifying the knowledge base.

A. Sound Recoding and Word detection:

The component responsibility is to accept input from a microphone and forward it to the future extraction module. Before converting the signal into suitable or desired form it identifies the segments of the sound containing words.

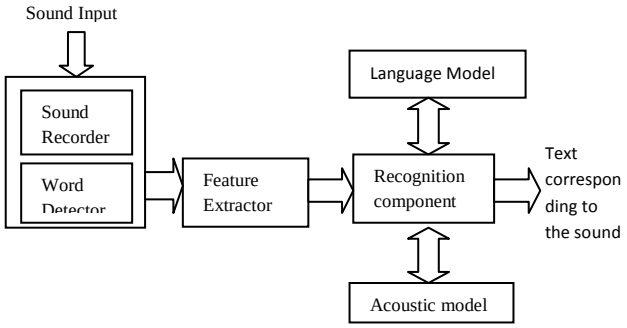

Fig. 2 Block Diagram of Training System

1) Sound Recorder:

The recorder takes input from the microphone and saves it or forwards it depending on the function invoked. Recorder supports changing of *sampling rate*, *channel* and *size of the sample*.

Initially, it is the job of Sound Reader class to take the input from the user. The sound reader class takes sampling rate, sample size and number of channels as parameters. The reader class takes care of converting the raw audio signal into wav format and stores it to a file.

2) Word Detector:

In speech recognition it is important to detect when a word is spoken. The system does detect the region of silence. Anything other than silence is considered to be a spoken word by the system. The system uses energy pattern present in the sound signal and zero crossing rate to detect the silent region. Taking both of them is important as only energy tends to miss some part of sound which is important.

B. Feature Extractor

Humans have a capacity of identifying different types of sounds (phones). Phones put in a particular order constitute a word. If we want a machine to identify the spoken word, it will have to differentiate between different kinds of sound the way the humans perceive it. The point to be noted in case of human is that although, one word spoken by different people produces different sound waves humans are able to identify the sound waves as same. On the other hand two sounds which are different are perceived as different by humans. The reasons being even when same phones or sounds are produces by different speakers they have common features. A good feature extractor should extract these features and use them for further analysis and processing.

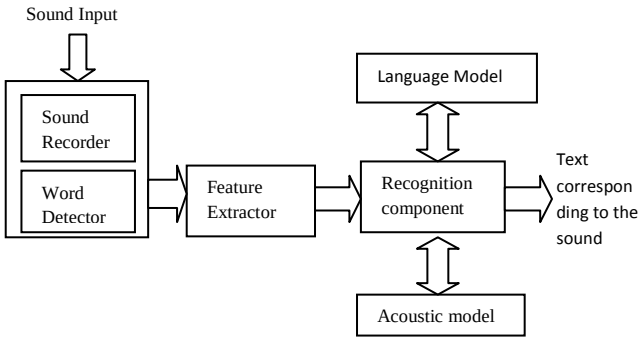

Fig. 2 Block Diagram of Training System

1) Windowing:

Features are periodically extracted. The time for which the signal is considered for processing is called a window and the data acquired in a window is as a frame. There are different types of windows which are used: Rectangular window, Bartlett window, Hamming window. The system uses Hamming window as it introduces the least amount of distortion. Impulse response of the hamming window is a raised cosine impulse. Transfer function of hamming window is

$$0.54 + 0.46\cos(n^{1/4}/m)$$

2) Temporal Feature:

Temporal features are easy to extract, simple and have easy physical interpretation. Temporal feature like average energy level, zero-crossing rate, root mean square, maximum amplitude etc can be extracted out as features.

3) Spectral Analysis:

Spectral analysis gives us quite a lot of information about the spoken phone. Time domain data is converted to Frequency domain by applying Fourier transform on it. This process gives us the special information. Spectral information is the energy levels at different frequencies in a given window. Thus features like frequency with maximum energy, distance between frequencies of maximum and minimum energies etc can be extracted.

Mel frequency cepstrum computation:

Mel frequency cepstrum computation (MFCC) is considered to be the best available approximation of humar ear. It is known that human ear is more sensitive to higher frequency. The spectral information can be converted to MFCC by passing the signals through band pass filters where higher frequencies are artificially boosted, and after then doing an inverse Digital Fourier Transform (DFT) on it. This result in higher frequencies being more prominent.

Feature extraction module is capable of producing different kinds of features from the sound input. The possible features that can be extracted are Energy, MFCC, their derivative coefficients and second order derivative coefficient.

4) Feature Vector Specification:

Vectors were generated at frame duration of 10 milli-seconds. Window used was hamming windows with duration of 25 milli-seconds. 12 MFCC and energy level are generated for each frame. These features can now be used for recognition or for training the HMM.

C. Knowledge Model:

For Speech recognition, the system needs to know how the words sound. For this we need to train the system. During the training, using the data given by the user, the system generates acoustic model and language model. These models are later used by the system to map a sound to a word or a phrase.

1) Acoustic Model:

Features that are extracted by the Feature Extraction module needs to be compared against a model to identify the sound that was produced as the world that was spoken. There are two kinds of Acoustic Models: Word Model and Phone Model

a) Word Model:

Word model are generally used to small vocabulary system. In this model the worlds are modeled as whole. Thus each words needs to be modeled separately. If we need to add support to recognize a new word, we will have to train the system for the word. In the recognition process, the sound is matched against each of the model to find the best match. This best match is assumed to be spoken word. Building a model for a word requires us to collect the sound files of the word from various users. These sound files are then used to train a HMM Model.

b) Phone Model:

In phone model instead of modeling the whole word, we model only parts of words generally phones. And the word itself is modeled as sequence of phone. The heard sound is now matched against the parts and parts are recognized. The recognized parts are put together to form a word.

For example the word “ek” is generated by combination of two phones A and k. This is generally useful when we need a large vocabulary system. Adding a new word in the vocabulary is easy as the sounds of phone are already know only possible sequence of phone for the word with it probability needs to be added to the system.

In both word acoustic model and phone acoustic model we need to model silence and filter words too. Filter words are the sounds that humans produce between two words. Both of these models can either be implemented using a Hidden Markov Model or a Neural Network. HMM is most widely used technique in automatic speech recognition system.

2) Language Model:

Although there are words that have similar sounding phone, humans generally do not find it difficult to recognize the word. This is mainly because they know the context, and also have a fairly good idea about what words or phrases can occur in the context. Providing this context to a speech recognition system is the purpose of language model. The language model specifies what are the valid words in the language and in what sequence they occur.

We will use a word acoustic model. The system has a model for each word that the system can recognize. The list of words can be considered as language model. While recognizing the system need to know where to locate the model for each word and what word the model corresponds to. This information is stored in a flat file called models in a directory called HMM’s.

When sound is given to the system to recognize, it compares each model with the word and finds out of the two model that most closely matches with it. The word corresponding to that HMM model is given as the output.

IV. HMM RECOGNITION AND TRAINING

Hidden Markov Model (HMM) is a state machine. The states of the model are represented as nodes and the transition are represented as edges. The difference in case of HMM is that the symbol does not uniquely identify a state.

The new state is determined by the transition probabilities from the current state to a candidate state.

A. HMM and speech recognition

While using HMM for recognition, we provide the occurrences to the model and it returns a number. This number is the probability with which the model could have produced the output (occurrences). In speech recognition occurrences are features vectors rather than just symbols. Hence for each occurrence, feature vector has a group of real numbers. Thus, what we need for speech recognition is a Continuous, Multi-dimensional HMM.

1) Recognition using HMM:

We need to recognize a word sing the existing models of words that we have. Sound recorder need to record the sound when it detects the presence of a word. This recorded sound is then passed through feature vector extractor model. The output of the above module is a list of feature taken every taken 10 msec. This feature is then passed to the Recognition module for recognition.

The list of all the words that the system is trained for and their corresponding models are given in a file called models present in the HMM's. All models corresponding of the words are then loaded in memory. The feature vectors generated by the feature vector generated module act as the list of observation for the recognition module. Probability of generation of the observation given a model, P (O|¸), is calculated for each of the model using find probability function. The word corresponding to the HMM, that gives the probability that is highest and is above the threshold, is considered to be spoken.

2) Training the Model:

Before we can recognize a word we need to train the system. Train command is used to train the system for a new word. The command takes at-least 3 parameters:

- No. of states the HMM model should have N.
- The size of the feature vector D.
- One or more filenames each containing a training set.

For generating an initial HMM we take the N equally placed observations (feature vector) from the first training set. Each one is used to train a separate state. After training the states have a mean vector which is size D and a variance matrix of size D * D containing all zeros. Then for each of the remaining observations, we find the Euclidean distances between it and the mean vector of the states. We assign an observation to the closest state for training. The state assigned to consecutive observations are tracked to find the transitional probabilities.

a) Segment K-means Algorithm:

This algorithm tries to modify the initial model so as to maximize P (O, I|¸), where O are the training sets used for training and I is a state sequence in the given HMM. The maximized (optimal) path for a training set is denoted by I*. Those observations that were assigned to a different state then the one in which they should be present according the optimal path are then moved to the state. This improves P (O, I*|¸), the model is evaluated again so with this changed assignments of observations. We do the above process iteratively till there are no more assignments needed.

b) Viterbi Algorithm:

This algorithm is useful identifying the best path that a signal can take in an HMM. To find the best path in a search problem Viterbi uses dynamic programming to reduce the search space [5].

V. CONCLUSION

This paper will demonstrate and an implementation of building an application which will give output in the form of text having user friendly interface and useful to day to day life for many disable users. We are giving the idea how we can create a Hindi Speech Recognizer and it works.

REFERENCES

[1] Ms. Anuja Jadhav, Prof. Arvind Patil / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com vol. 2, Issue 2, Mar-Apr 2012, pp.1126-1128, "A smart Testing system for Android Mobile users".

[2] IOSR Journal of Engineering Mar. 2012, Vol. 2(3) pp. 420-423 ISSN: 2250-3021 www.ijera.org 420| "Android Speech to Text Converter for SMS Application" Ms. Anuja Jadhav, Prof. Arvind Patil.

[3] N. Rajput M. Kumar and A. Verma. A large–vocabulary continuous system for Hindi. IBM Journal for Research and Development.

[4] Jagriti Chand, "SMS to Text Convertor in android Mobiles", Application using speech"| International Journal in computer Science & Electronics, vol. 1 Issue 1, Ref. Id: aijcse2005.

[5] Mr. Ripul Gupta," Speech Recognition for Hindi", Department of Computer science and Engineering, Indian Institute of Technology, Bombay.

[6] International Journal of Information and communication Engineering 6:1 2010 "The main Principles of Text-to-Speech Synthesis System", K.R.Aida- Zade, C.Ardil and A.M.Sharifova