Scientific Journal of Impact Factor(SJIF): 3.134

e-ISSN(O): 2348-4470
p-ISSN(P): 2348-6406

International Journal of Advance Engineering and Research Development

Volume 2,Issue 3, March -2015

SPEECH SYNTHESIS USING VOICE COMMAND IDENTIFICATION

Tejashree M. Shinde[1], V.U.Deshmukh[2], P.K.Kadbe[3]

[1]P.G.Scholar EnTC Dept. VPCOE,Baramati,Pune,Maharashtra ,India.
[2] VPCOE,Baramati,Pune,Maharashtra ,India.
[3]VPCOE,Baramati,Pune,Maharashtra ,India.

Abstract :*A text to speech (TTS) converter is a computer based system that can read input text aloud automatically, the input text may be introduced by a computer input stream or it may be scanned input submitted to an Optical character recognition (OCR) engine. A Text to Speech Converter can be implemented by both hardware and software. A very fast improvement is found in this field over the couple of decades and lots of high quality TTS systems are now introduced for commercial use. This paper describes a Text to Speech (TTS)Synthesizer which can be operated by blind or visually impaired people by using a voice commands. Here PocketSphinx voice recognition algorithm is used for converting voice commands into text and FLITE Text to Speech synthesizer engine is used to convert the input text into speech. This system is more user friendly for those people who are very much dependent on the speech information for getting any knowledge.*

Keywords- *Text to Speech(TTS) System , FLITE system, PocketSphinx Algorithm, cepstrum, CMU Sphinx.*

I. INTRODUCTION

Speech is the primary means for humans to communicate with one another. The automatic conversion of written text into human speech would be a very useful technology with many applications. Speech synthesis is the automatic generation of speech waveforms. The text-to-speech (TTS) synthesis procedure consists of two main phases. The first is text analysis, in which the input text is translated into a phonetic or some other linguistic representation, and the second phase is the generation of speech waveforms, where phonetic and prosodic information produces acoustic output. The Speech Recognition System is used to operate Text to Speech Synthesizer.

Speech recognition is the translation of a raw audio signal into text as shown in Figure with accuracy as high as possible. The recognized text can be used further in a system or presented directly to the user as text. Speech recognition and Text-To-Speech have been in research for many years. Speech recognition continues to be an important research topic to increase the accuracy, because of variations in the environment, context, and speaker .

Figure 1:Basic idea of Speech recognition

The main aim of Speech Recognition is that creating artificial models of human verbal communication, so that humans and machines can communicate with each other.

1.1 Speech Recognition systems:

The speech recognition systems can be divided into two main classes. The first one is the command and control type of system, where the system can interpret a predetermined, mostly very small set of words or sentences(isolated words). This controlling systems type of recognition is often used where specific words or phrases are used to start or activate different things in the system.The command and control-type of system is the interactive voice response systems. These are mostly used in call centers the caller can give the voice request and get the information they need.

The another type of system is dictation systems. This system recognize all (as many as possible) words that user is speaking from the continuous speech. These systems need a language model which serves same as the limited set of sentences in command and control types of systems.

1.2 Requirements for speech recognition:

A speech recognizer require an acoustic model and preferably a language model to work.

The acoustic model is the representation of different raw audio input signals and the mapping of these signals to phonemes. The phonemes can be mapped into the possible words, by having a dictionary which consist of a mapping between phonemes and written words. The language model contains rules of how the language to be recognized is organized, which basically consists of rules of how the words in the language can be ordered. Language models are used to limit the search space of the recognition, i.e. the most likely word sequences get highest priority. A very simple

language model can contain simple and strict rules of sequences of word which are recognized. Simple models are mostly used for the command and control types of speech recognition systems, whereas statistical models are normally used for the dictation types of systems. By combination of the acoustic model and the language model, through the extraction of phonemes from the audio and using the language model to produce the most likely phrase from those phonemes, speech recognition is in work, where the feature vectors contain recognized features of the raw audio signal. Feature vectors in a certain order can be translated as phonemes.

1.3 Difficulties in speech recognition:

Speech recognition is complicated by many factors, some external ones that are hard to address as well as some that are possible to improve. External factors, such as audio quality and noise can have a huge impact on the quality of the speech recognition. Noise, for example city background noise or audience sounds, sometimes can be loud compared to the relevant audio. This causes the sound waves to be altered which causes the recognized phonemes to be different, which will detect falsely words. Noise can also be the result from bad recording equipment. The size of the vocabulary will affect speed and accuracy of the recognition. A larger dictionary means that more words need to be considered for the recognition. If a system is meant to do limited speech recognition, like command and control systems, the vocabulary can often be limited to a very small set of words, ultimately the accuracy and speed can also be improved.

The quality of the spoken words is also very important. Words can for example be spoken in different speeds, in different volumes, different pitches,different accents, which makes the signal processing more difficult. In an ideal world, when it comes to speech recognition, everyone should talk in exactly the same way and use the same build-up of sentences and equipment that would yield perfect recording quality with no noise.

II. METHODOLOGY

This system is designed to be operated by using three commands ‘one’, ‘two’, and ‘four’. When user give any of these command then the Pocketsphinx Algorithm detect the command and convert that voice command into text. Then the respective text file for the command is selected and this text is converted into speech by using Flite Text to Speech Algorithm.

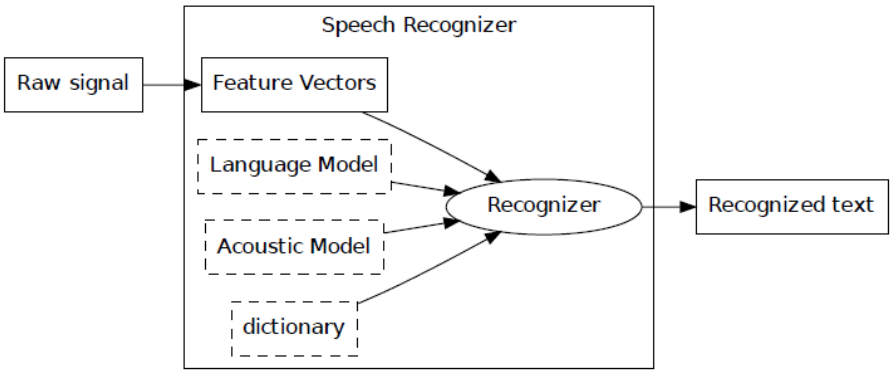

Figure 2:Block Diagram of Speech Recognizer

2.1 Fundamental components of Speech Recognizer:

2.1.1 Feature Extraction

The raw audio signal is converted to a spectrum and a cepstrum, often using the Fast Fourier Transform(FFT) algorithm. This algorithm essentially converts the audio signal from the "time domain" to the "frequency domain". From the cepstrum, feature numbers are extracted and stored into feature vectors. The Fourier Transform is not used directly but instead the Perceptual Linear Prediction or Mel Frequency Cepstral is used. Both of these however work by calculating the feature numbers (vectors) from the audio signal in chunks. Each phoneme in a language consists of one or multiple feature vectors after one another. The feature vectors are stored in the acoustic model that represents different phonemes in a language. Using statistics, it is possible to produce a list of feature vectors from an input, and feature vectors can then be mapped to phonemes and words.

2.1.2 Statistical Models

The most common class of speech recognition makes use of statistical models, which means that the recognition process makes use of probabilistic methods.Having a statistical database over how different phonemes are normally structured and how a language is structured, it is possible to analyze the input signal to determine the most likely sequence of

feature vectors and therefore also phonemes and words. This is carried out by using previous information about the input signal to determine the next state.

2.1.3 Acoustic models

An acoustic model is a representation of the pronunciation of phonemes of a language. The English language has around 40 phonemes. A lexicon or a dictionary describes the vocabulary items and a representation of the pronunciation of the items, where each item is usually a word and the pronunciation is often the sequence of phonemes that the word consists of. The dictionary can hold multiple entries for one word, meaning that each word can be pronounced different ways. The phonemes themselves are derived from statistical data, which in turn is generated from a corpus(a text bank with associated spoken audio). VoxForge is an example that provides such a corpus. The corpus is used to create a statistical representation of the phonemes in the language. These statistical representations are very commonly stored as Hidden Markov Models.The accuracy of the acoustic model is dependent of the consistency of the pronunciations listed in the lexicon.This makes sense, since having a single phoneme representing two different sounds would not cause a very accurate recognition.

2.1.4 Language models

A language model describes the legal sentences in a language, or how words in a language can be ordered to be part of the language. These are used to limit the search space of the recognizer. The recognizer would have to consider every possible word combination in the vocabulary without the language model.N-grams are very common in speech recognition and consist of all legal word sequences of length N which are allowed in the language. Each N-gram contains a probability that represents the probability that the N words in the N-gram are used in that order in the language.

Grammar: A grammar is a type of language model, and consists of rules in which words can occur and is often implemented as a context-free grammar. A grammar can make use of word classifications, in which each word in the dictionary is classified as a verb, noun, prepositions and so on. An example grammar can for example contain the word order 'noun verb preposition noun'. A grammar can be more simplistic than that and can in a very simple form specify the specific words that can be recognized,and can also describe very complex languages.

Language models that use probabilities for word occurrences use estimates of how often those word combinations occur in a corpus. From the corpus, the frequency of each word and each N-gram is calculated, from which it is an easy task to calculate the probability of each word and each N-gram.

2.2 Speech Recognition Systems and Libraries

There are some libraries available that are reasonable for the evaluation. The main requirement for the libraries is that they should be able to recognize large vocabulary continuous speech. The ones used in the evaluation of speech recognizer are CMU Sphinx, Microsoft SAPI and Dragon NaturallySpeaking. A valuable resource for the evaluation is VoxForge. It provides a database with audio files and with corresponding text, which makes it possible to generate acoustic models to be used with a recognizer. VoxForge provides acoustic models for CMU Sphinx, Julius and HTK, which can be downloaded from the VoxForge web page. Most Acoustic Models used by Open Source Speech Recognition engines are closed source, according to VoxForge. These closed source acoustic models do not provide access to the speech audio (the source) used to create the Acoustic Model, since they often are licensed. VoxForge is a good starting point when generating a custom acoustic model.

2.2.1 CMU Sphinx

CMU Sphinx is an open source speech recognition toolkit, containing both a speech recognition engine and an acoustic model trainer. The speech recognition engine exists in two versions, one written in C (PocketSphinx, Sphinx3) and one written in Java (Sphinx4). The one written in C is chosen for the evaluation. The CMU Sphinx toolkit was developed at Carnegie Mellon University as the name suggests.CMU Sphinx also contains acoustic model training tools. This can be used to adapt an acoustic model to improve accuracy or to create an entirely new acoustic model, for example for a new language.

CMU Sphinx has the advantage of being available on Linux, Windows and Mac OS X. There exist acoustic models, language models and dictionaries for this library that are available for download. It is also possible to create and use other models and dictionaries. CMU Sphinx had a couple of candidate test libraries, namely PocketSphinx or Sphinx4. Sphinx4 is written in Java and PocketSphinx is written in C, they are however said to be performing equally well. PocketSphinx was chosen since it would fit better into the embedded system. For PocketSphinx, a sample program was written in C.

2.3 Flite Algorithm:-

Flite is a small,but fast run-time synthesis library suitable for embedded systems and servers. Flite is designed as another substitute run-time synthesis platform for Festival where the speed and size are important. The Voices built by the FestVox process may be gathered into efficient representations that can be linked against Flite to produce a complete text to speech synthesizer. The Flite library is very much faster and much smaller compared with equivalent Festival system.

Synthesized speech can be produced by concatenating pieces of recorded speech which are stored in a database. These systems differ in the size of the stored speech units; a system that stores phones (or diphones) provides the largest output range, but may give low clarity. For specific applications, the storage of entire words or sentences useful for high-quality

output. On other hand, a synthesizer can also incorporate a model of the vocal tract and other human voice characteristics to create a completely "synthetic" voice output. The quality of a speech synthesizer is decided by its naturalness to the human voice and by its ability to be understood clearly. An intelligible text-to-speech program very much helpful for people with visual impairments or reading disabilities to listen to the written works on a home computer.

Fundamental Phases of TTS System
2.3.1.Text Analysis and Text Detection
2.3.2Text Normalization and Text Linearization
2.3.3Phonetic Analysis
2.3.4Prosodic Modeling and Intonation
2.3.5Acoustic Processing

2.3.1.Text Analysis and Text Detection
The Text Analysis is preprocessing part of TTS System which analyses the input text and organize into manageable list of words. It consists of abbreviations, numbers, acronyms and idiomatic and transforms them into full text. Text detection is localizing the text areas from any kind of printed documents.

2.3.2Text Normalization and Text Linearization
Text Normalization is the process of conversion of text to pronounceable form.
Mostly the text normalization process is done for converting all letters of lowercase or upper case, accent mark , to remove punctuations, stop words and other diacritics from letters . The main phases of Text Normalization are
Number converter, Abbreviation converter, Acronym converter, Word segmentation.

2.3.3Phonetic Analysis
The Phonetic Analysis process converts the orthographical symbols into phonological ones using a phonetic alphabet. It is basically known as “grapheme-to-phoneme” conversion.
Phoneme Set (English)
Vowels (19) : /a/, /ae/, /ar/, /e/, /i/, /ie/, /o/, /oe/,/air/ /oi/, /ow/, /or/, /u/, /ur/,/oo/, /ue/, /uh/, /w/,/ee/.
Consonants (25) : /gz/, /c/,/k/, /ch/, /d/, /f/, /g/, /h/, /l/, /m/, /ng/, /p/, /kw/,/n/, /r/, /s/,/j/, /sh/, /t/, /th/, /th/ , /v/, /y/, /z/, /b/,/zh//ks/.
Examples:
- */air/ : square, fear.*
- */ow/ :Close, house.*
- */ks/gz/ : Mix, exist.*

2.3.4Prosodic Modeling and Intonation
The prosody is the combination of stress pattern, intonation and rhythm in a speech. The prosodic modeling describes the speaker’s emotions. Intonation is a variation of speech while speaking. Pitch is used as an intonation in all languages, to express happiness, to raise a question etc.

2.3.5Acoustic Processing
The speech should be spoken according to the voice characteristics of a person.Synthesized speech can be produced by concatenating pieces of recorded speech which are stored in a database. These systems differ in the size of the stored speech units; a system that stores phones (or diphones) provides the largest output range, but may give low clarity. For specific applications, the storage of entire words or sentences useful for high-quality output. On other hand, a synthesizer can also incorporate a model of the vocal tract and other human voice characteristics to create a completely "synthetic" voice output.The quality of a speech synthesizer is decided by its naturalness to the human voice and by its ability to be understood clearly. An intelligible text-to-speech program very much helpful for people with visual impairments or reading disabilities to listen to the written works on a home computer.

III. RESULT

Speech recognition System detect the given command and compare that command in its vocabulary and then convert that file into speech at the output.

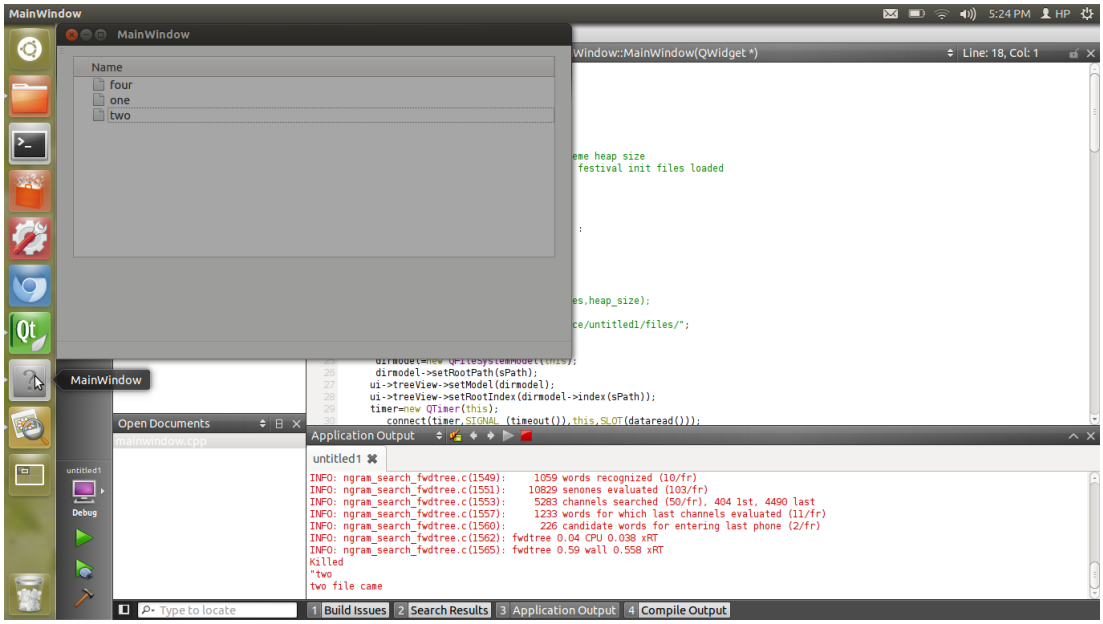

Figure3: Speech Output for file 'two'

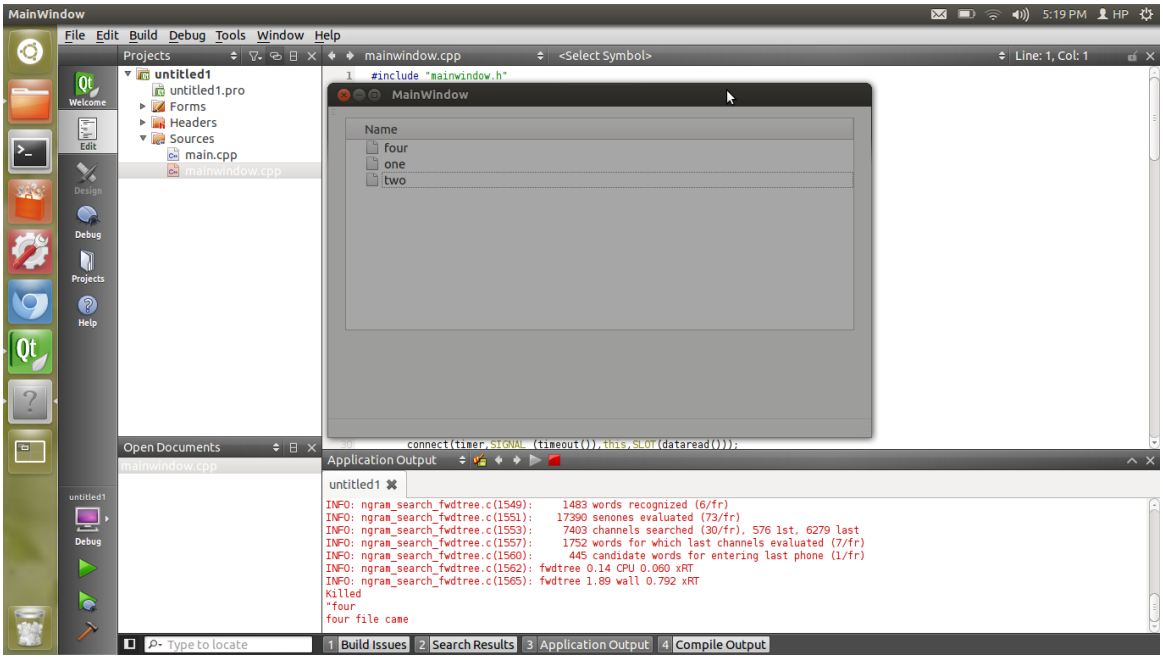

Figure 4:Speech output for File 'Four'

IV.CONCLUSION

There are many text to speech (TTS) converters available in the market and also much improvement is going on in the research area to make the speech more effective. A Flite-based synthesizer runs on multiple platforms. The Run-time memory requirements for Flite are less than twice the size of the largest waveform built. PocketSphinx is the best candidate for the intended system on the basis that it is open-source, free and would be a better match to the system.
By modifying the language model and dictionary it is also possible to improve the WER for PocketSphinx to a value of 39.5%.Pocket-Sphinx is written in C and has support for Linux. Customizing the language model and dictionary it is possible to achieve results that performed better than any other speech recognition software.

V.REFERENCES

[1]Wiqas Ghai , Navdeep Singh Literature Review on Automatic Speech Recognition"International Journal of Computer Applications (0975 – 8887) Volume 41– No.8, March 2012.

[2]Tobias Nilsson "Speech Recognition Software and Vidispine" Umea University Department of Computing Science SE-901 87 UMEA SWEDEN April 2, 2013.

[3]Alan W Black and Kevin A. Lenzo; Flite: a small fast run-time synthesis engine, Carnegie Mellon University.

[4]D.Sasirekha, E.Chandra.Text to Speech: A Simple Tutorial, International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-1, March 2012.

[5]Atsushi Imai, Naoyuki Tazawa, Tohru Takagi, Toshiaki Tanaka and Tohru Ifukube.A New Touchscreen Application to Retrieve Speech Information Efficiently,IEEE Transactions on Consumer Electronics, Vol. 59, No. 1, February 2013.

[6]Alan O'Cinneide, David Dorran, Mikel Gainza, "A Brief Introduction to Speech Synthesis and Voice Modification", Dundalk Institute of Tchnology, Ireland. November 2007.

