INTERACTION MODULE FOR HUMANOID ROBOT

A Project Report

submitted by

in partial fulfilment of the requirements
for the award of the degree of

BACHELOR OF TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
COLLEGE OF ENGINEERING,TRIVANDRUM.
May 2019

THESIS CERTIFICATE

This is to certify that the thesis entitled **INTERACTION MODULE FOR HUMANOID ROBOT**, submitted by submitted by **Gokul p ,Sreehari k , Ashiq pt, Sandra Mariya Jose** to the APJ Abdul Kalam Technological University , for the award of the degree of **Bachelors of Technology** is a bona fide record of the research work carried out by them under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Prof. Rajasree R	**Prof. Vipin Vasu A V**	**Dr. Salim A**
Assistant Professor	Associate Professor	Professor
(Project Guide)	(Project Coordinator)	(Head of Department)

ACKNOWLEDGEMENTS

Firstly, we would like to thank the almighty for giving us the wisdom and grace for presenting our project design.

With a profound sense of gratitude, we would like to express our heartfelt thanks to our guide **Prof. Rajasree R**, Assistant Professor, Department of Computer Science and Engineering for her expert guidance, co-operation and immense encouragement in pursuing this project presentation.

We are very much thankful to **Dr. Salim A**, Head of Department and
Prof. Vipin Vasu A V, Associate Professor of Department of Computer Science and Engineering, our guide for providing necessary facilities and his sincere co-operation.

Our sincere thanks is extended to all the teachers of the department of CSE and to all our friends for their help and support.

Gokul P

Sreeahri K

Ashiq Pt

Sandra Mariya Jose

ABSTRACT

Speech recognition and related tasks for many languages are gaining more importance in the present scenario. In the case of Malayalam language, recognizing speech is a tiresome task. This report presents a speech recognition and response phase of an Interaction module in Malayalam language. The system is speaker independent with limited vocabulary and considers only isolated words. Interaction module will listen to the audio queries of the user and respond to them with audio output. CMUSphinx open source tool is used for the creation of speech recognizer and custom mapping algorithm is used for mapping output audio with the recognized text.

Keywords: Interaction module, Speech recognizer, CMUSphinx,Output audio mapping.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Speech is the primary form of communication and it is considered effective for exchange of information.Verbal form of communication is more efficient compared to other written methods. Speech provides an interface for humans to interact with machines.

Malayalam is one of the four major languages belonging to the Dravidian family of languages with a rich literary tradition. Malayalam is spoken by around 37 million people and it is the official language of the state of Kerala,India. Malayalam language is very phonetic in nature and also has a canonical word order of Subject-Object-Verb (SOV). Malayalam consists of 16 vowels and 37 consonants,though this has been to slight modifications through ages.
In terms research and development Robotics is one of the most advancing areas.Path breaking inventions are happening in this field on daily basis. But still there is no robot which can understand Malayalam commands. So we are trying to make a Malayalam interaction module for the robots.

A good offline speech recognizer is still not available in Malayalam language. So making such a speech recognizer will be very usefull in various applications. In this project the interaction module will be handling Malayalam language.All the input and output will be in Malayalam language. Interaction module will be consisting two parts one speech recognizer and an output audio mapper. CMUSphinx open source tool is used for the creation of the speech recognizer and custom mapping algorithm is used for mapping output audio with the recognized text.

CHAPTER 2

PROBLEM DEFINITION

Development of an interaction module ,which can understand and reply to queries in Malayalam language. An user who interacts with the application which has the interaction module, will ask query in Malayalam. The interaction module takes an audio input and converts this audio file to text using speech recognition system developed. The interaction module will select appropriate audio response and output it.

CHAPTER 3

REQUIREMENTS

3.1 Functionality Requirements

Input to the module will be any Malayalam query audio in .wav file extension. .wav file is a considered because it is will give compressed file without any loss. The speech recognizer process the input audio and output the corresponding text. This text is given to the speech synthesizer. It will map the text recognized to corresponding output. This mapped audio file is outputted as response to the input query. The Use case diagram of the module is presented in the figure 4.1

3.2 Quality requirements

- Data collection is very time consuming process.
- Training requires high computational power.
- Large amount data required for training.
- Recognition in noisy environment is difficult.
- Over fitting may occur for repeatedly trained words.

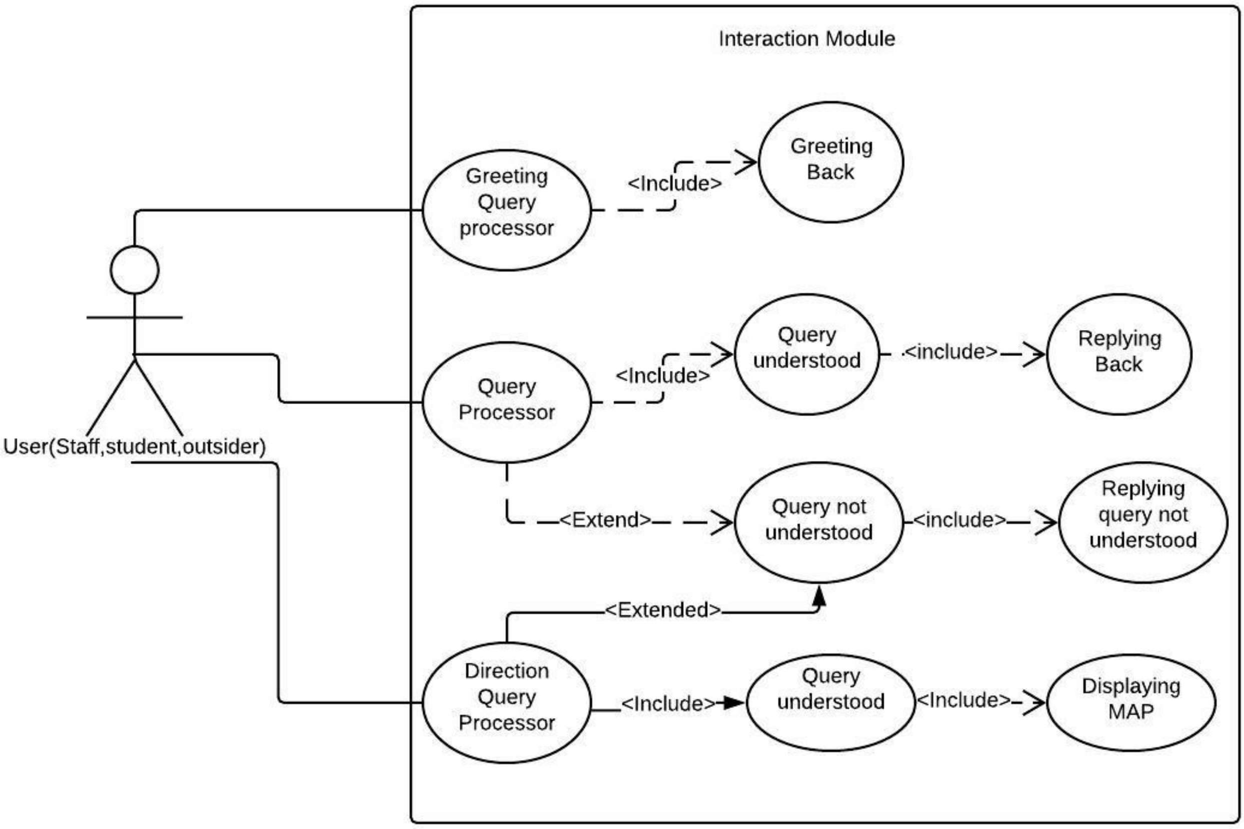


Figure 3.1: Use case diagram of the interaction module

CHAPTER 4

METHODOLOGY

4.1 CMU Sphinx

CMU Sphinx is the proposed platform for Speech recognition.It is an open source speech recognition system and has many versions which meets different environments in which the system is employed.Sphinx train is used as the acoustical model trainer. It is a statistical acoustic model trainer using Hidden Markov Models

4.2 Data Collection

25 Custom sentences required for the interaction module is made. Audio files to be collected are required to be in 16khz and the fles are in .wav format and its bit rate is 256 kbps. 64 audio recordings of each sentence in the required format is collected. ie total of 1794 audio files are collected that correspond to 1 hour 12 minutes of data. Transcribed files along with corresponding field ids were created for each sentences.

4.3 Architecture

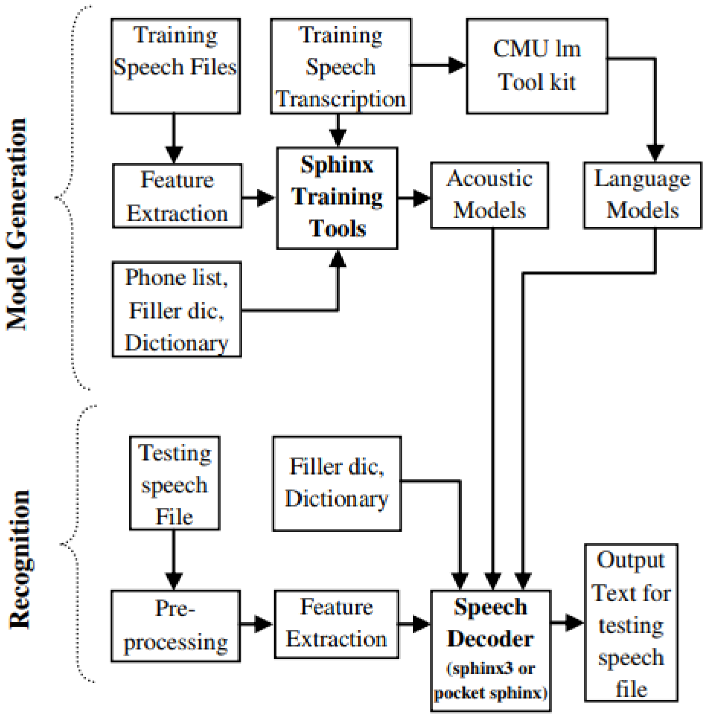
The proposed system is divided into:

- Speech recognition phase.
- Output speech mapping phase.

4.3.1 Speech recognition phase.

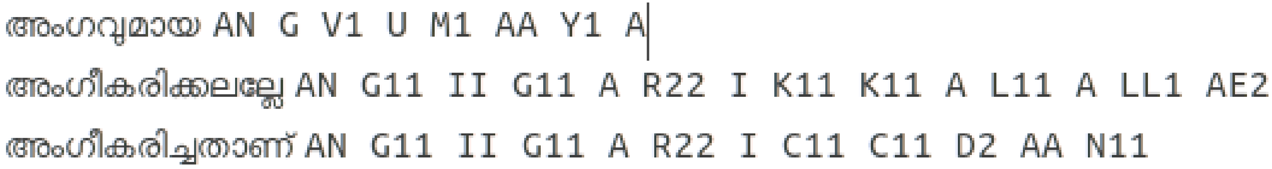
In speech recognition phase an audio input is taken from microphone and fed to the speech decoder. The speech decoder will do STT conversion and outputs Malayalam text data of the input speech.These are the items required for speech recognition.

- Phonetic Dictionary : A phonetic dictionary provides the system with a mapping of vocabulary words to sequences of phonemes.
- Language Model : A Language Model is a file used by a Speech Recognition Engine to recognize speech. It help to guide and constrain the search among alternative word hypotheses during decoding. It contains a large list of words and their probability of occurrence.
- Acoustic Model : Acoustic Model represent relationship between audio and phonemes.



4.3.2 Phonetic Dictionary

A phonetic dictionary provides the system with a mapping of vocabulary words to sequences of phonemes. It might look like this:



```
അംഗവുമായ AN G V1 U M1 AA Y1 A
അംഗീകരിക്കലല്ലേ AN G11 II G11 A R22 I K11 K11 A L11 A LL1 AE2
അംഗീകരിച്ചതാണ് AN G11 II G11 A R22 I C11 C11 D2 AA N11
```

A dictionary should contain all the words for the recognizer to be able to recognize them. The recognizer looks for a word in both the dictionary and the language model.Phonetic representation of all the required words are created and added to the dictionary

4.3.3 Language Model

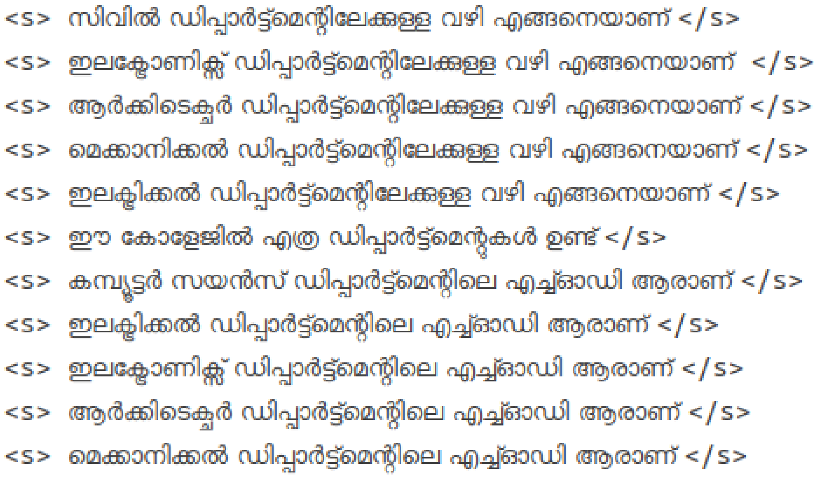
The CMU language modeling toolkit is used for developing laguage model.There are three different models in which a language model can be stored and loaded.

- Text ARPA format
- Binary BIN format
- Binary DMP format

The ARPA format can be edited but it takes more spac. ARPA files have an .lm extension. Binary formatshave a .lm.bin extension and take significantly less space and load faster. ARPA model is trained with CMUCLMTK.

The language model can be developed by following the process given below:

- Training transcript text is used for the creating language model.
- The language model toolkit expects its input to be in the form of normalized text files, with utterances delimited by ”¡s¿” and ”¡/s¿” tags. Result will be:



```
<s>  സിവിൽ ഡിപ്പാർട്ട്മെന്റിലേക്കുള്ള വഴി എങ്ങനെയാണ് </s>
<s>  ഇലക്ട്രോണിക്സ് ഡിപ്പാർട്ട്മെന്റിലേക്കുള്ള വഴി എങ്ങനെയാണ്  </s>
<s>  ആർക്കിടെക്ചർ ഡിപ്പാർട്ട്മെന്റിലേക്കുള്ള വഴി എങ്ങനെയാണ് </s>
<s>  മെക്കാനിക്കൽ ഡിപ്പാർട്ട്മെന്റിലേക്കുള്ള വഴി എങ്ങനെയാണ് </s>
<s>  ഇലക്ട്രിക്കൽ ഡിപ്പാർട്ട്മെന്റിലേക്കുള്ള വഴി എങ്ങനെയാണ് </s>
<s>  ഈ കോളേജിൽ എത്ര ഡിപ്പാർട്ട്മെന്റുകൾ ഉണ്ട് </s>
<s>  കമ്പ്യൂട്ടർ സയൻസ് ഡിപ്പാർട്ട്മെന്റിലെ എച്ച്ഓഡി ആരാണ് </s>
<s>  ഇലക്ട്രിക്കൽ ഡിപ്പാർട്ട്മെന്റിലെ എച്ച്ഓഡി ആരാണ് </s>
<s>  ഇലക്ട്രോണിക്സ് ഡിപ്പാർട്ട്മെന്റിലെ എച്ച്ഓഡി ആരാണ് </s>
<s>  ആർക്കിടെക്ചർ ഡിപ്പാർട്ട്മെന്റിലെ എച്ച്ഓഡി ആരാണ് </s>
<s>  മെക്കാനിക്കൽ ഡിപ്പാർട്ട്മെന്റിലെ എച്ച്ഓഡി ആരാണ് </s>
```

- Vocabulary file is generated using ”text2wfreq” and ”wfreq2vocab” scripts provided by the Sphinx
- Finally, using previously generated vocab file idngram file is created and then with that ”.lm” file is created. For all these steps sphinx provides some commands.

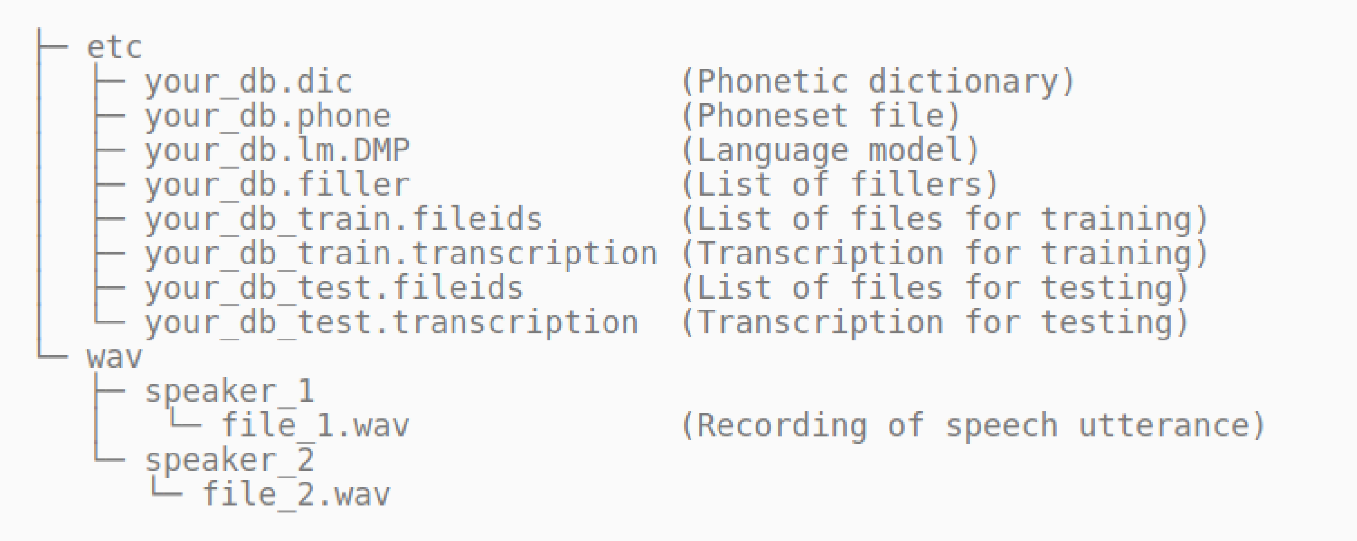
4.3.4 Acoustic model

Acoustic model is developed using sphinx train package . Acoustic models capture the characteristics of the basic recognition units. Sphinx train has the training algorithms implemented so that we can configure the system and give the required data.

Data preparation for acoustic model

Sphinx needs the collected data to be prepared like this database structure.

- The your_db_train.fileids and your_db_test.fileids files are text files which list the names of the recordings one by one.
- The your_db_train.transcription and your_db_test.transcription files are text files listing the transcription for each audio file.



```
├─ etc
│  ├─ your_db.dic                 (Phonetic dictionary)
│  ├─ your_db.phone               (Phoneset file)
│  ├─ your_db.lm.DMP              (Language model)
│  ├─ your_db.filler              (List of fillers)
│  ├─ your_db_train.fileids       (List of files for training)
│  ├─ your_db_train.transcription (Transcription for training)
│  ├─ your_db_test.fileids        (List of files for testing)
│  └─ your_db_test.transcription  (Transcription for testing)
└─ wav
   ├─ speaker_1
   │   └─ file_1.wav              (Recording of speech utterance)
   └─ speaker_2
      └─ file_2.wav
```

- Phonetic Dictionary (your_db.dict): should have one line per word with the word following the phonetic transcription.
- Phoneset file (your_db.phone): should have one phone per line. The number of phones should match the phones used in the dictionary plus the special SIL phone for silence.
- Filler dictionary (your_db.filler): contains filler phones (not-covered by language model non-linguistic sounds like breath, hmm or laugh)

After setting up all these files sphinxtrain is used to train the acoustic model using this database. CMUSphinx supports different types of the acoustic models: continuous, semi-continuous and phonetically tied (PTM). Here PTM acoustic model is used because it requires less computations and gives accuracy close to continuous model.These models are given to sphinx decoder to do the decoding.The decoder will provide the transcription of given audio.

4.3.5 Output speech mapping phase

After Speech recognition, all the words of the speech is extracted using string manipulation to get text data of the speech. Hierarchical directory tree is used to map the keywords to the output file. It is done by identifying multiple keyword,

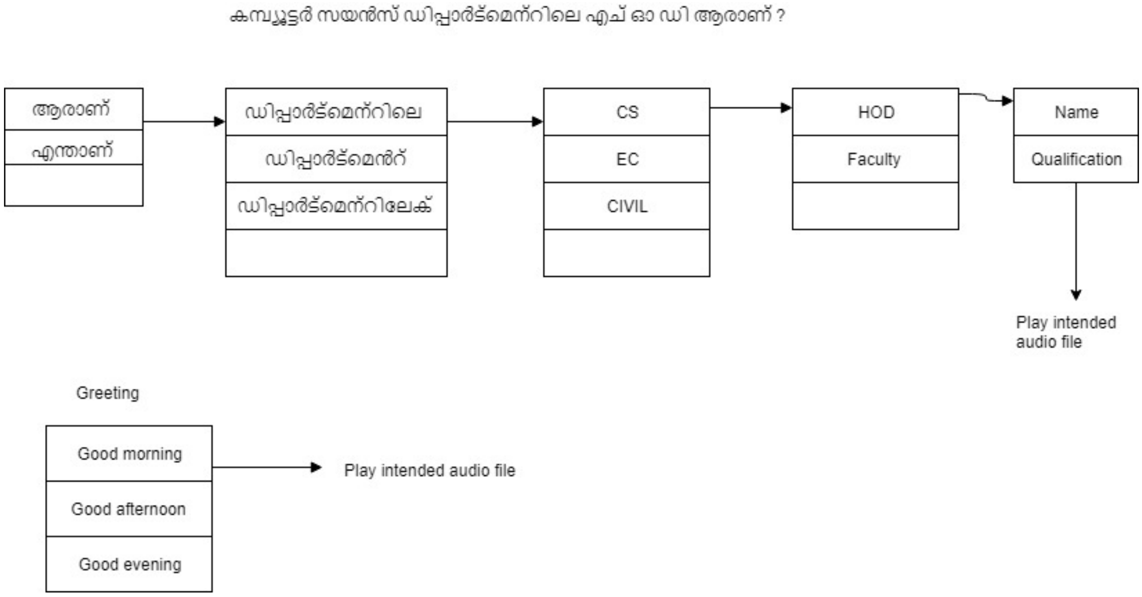


Figure 4.1: Mapping of input text to output audio file example

and mapping sentence from the speech recognition phase into a valid question. Since the speech synthesis part is very difficult Malayalam language considered. Almost all the responses needed for our project deploying context are recorded. These prerecorded audio is selected according to the input text. This mapping is done by considering all the words from the input text. At the end of the index traversal the audio file of the corresponding query is played. Otherwise not understood (in Malayalam) audio is outputted.

CHAPTER 5

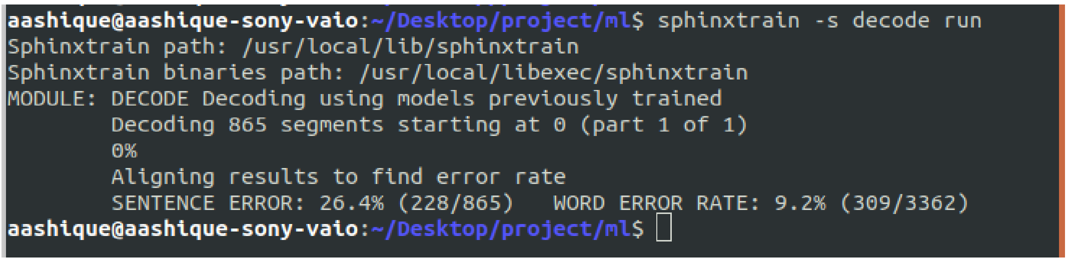
RESULTS OF DISCUSSION

5.1 Testing/Training/Dataset Accuracy status report

To get accurate speech recognizer we will be needing large amount of transcribed malayalam audio file. This is used with sphinx tools to generate the recognizer. After creating the speech recognizer, it is important to understand whether the accuracy is same as expected.First and foremost collect a database of test samples and measure the recognition accuracy. For that utterances are needed to be dumped into wav files. Then write a reference text and use the decoder to decode it. Finally calculate the Word Error Rate (WER) using the word align.pl tool from Sphinxtrain.The size of the test database depends on the accuracy but usually its sufficient to have 30 minutes of transcribed audio to test recognizer accuracy reliably.

5.2 Accuracy Obtained

Accuracy testing is done with the word align tool provided by the Sphinx. For the tool, a directory containing all the testing wav files and its transcriptions are needed. The tool will check actual transcription with the recognition engine's transcription. For our model 92 percent accuracy is obtained for word recognition and an accuracy of 74 percent is obtained for sentences.



```
aashique@aashique-sony-vaio:~/Desktop/project/ml$ sphinxtrain -s decode run
Sphinxtrain path: /usr/local/lib/sphinxtrain
Sphinxtrain binaries path: /usr/local/libexec/sphinxtrain
MODULE: DECODE Decoding using models previously trained
        Decoding 865 segments starting at 0 (part 1 of 1)
        0%
        Aligning results to find error rate
        SENTENCE ERROR: 26.4% (228/865)   WORD ERROR RATE: 9.2% (309/3362)
aashique@aashique-sony-vaio:~/Desktop/project/ml$ 
```

5.3 Git Log

commit 7af998dbab1c377a1c34602a18135faaea0dacc2 (HEAD -¿ master, origin/master, origin/HEAD) Author: Aashique ¡AashiqueBasheerPt@outlook.com¿ Date: Tue Mar 19 21:57:38 2019 +0530

limited case language model for humanoid project https://www.overleaf.com/project/5bf9026cf lismited case language model for humanoid project

commit e7ea427deb67bb5eb4cd8edff3ffd4cc1104bb45 Author: Aashique ¡AashiqueBasheerPt@outlook.com¿ Date: Tue Mar 19 21:53:04 2019 +0530

Add files via upload

language model made with the help of sreenaths language model

commit 521dd97441a8c1f8d34bc12b2623e52f007d7591 Author: GOKUL P ¡gokulpulikkal@rediff.com¿ Date: Tue Mar 19 21:09:18 2019 +0530

This is an acoustic model for Malayalam. This is made with the help of sreenath's existing acoustic model for malayalam. :...skipping... commit 7af998dbab1c377a1c34602 (HEAD -¿ master, origin/master, origin/HEAD) Author: Aashique ¡AashiqueBasheerPt@outlook.com¿ Date: Tue Mar 19 21:57:38 2019 +0530

limited case language model for humanoid project lismited case language model for humanoid project

commit e7ea427deb67bb5eb4cd8edff3ffd4cc1104bb45 Author: Aashique ¡AashiqueBasheerPt@outlook.com¿ Date: Tue Mar 19 21:53:04 2019 +0530

Add files via upload

language model made with the help of sreenaths language model

commit 521dd97441a8c1f8d34bc12b2623e52f007d7591 Author: GOKUL P ¡gokulpulikkal@rediff.com¿ Date: Tue Mar 19 21:09:18 2019 +0530

This is an acoustic model for Malayalam. This is made with the help of sreenath's existing acoustic model for malayalam.

commit e980655ac919f5e6d404b0a97441ab1585f783d6 Author: sandramariya ¡sandramariajose97@gmail.com¿ Date: Tue Mar 19 18:04:34 2019 +0530

This is an english speech recognition for limited number of commands. The language model is made with the help of sphinx website

commit c9722169de206fe4d657f81dcf16af14a02731da Author: Sreehari ¡sreeharik@cet.ac.in¿ Date: Tue Mar 19 16:35:47 2019 +0530

adding files

CHAPTER 6

RELATED WORKS

Malayalam speech recognition is a tiresome task.Therefore only limited number of works are done on malayalam speech recognition.

AAM Abusharisha et al.[5] developed a speech recognition system on all English digits from (Zero through Nine).Experiments were conducted for isolated words speech recognition and the continuous speech recognition.They employed MFCC and HMM for developing the system. The authors acheived 99.5% for multi-speaker mode and 79.5% accuracy for speaker-independent mode for isolated words and for continuous speech recogntion 72.5% for multi-speaker mode and 56.25% speaker-independent mode respectively.

Bassam A. Q. Al-Qatab , Raja N. Ainon [6] discussed the development of Arabic automatic speech recognition engine. Hidden Markov Model Toolkit was used to develop the system. They employed Mel Frequency Cepstral Coefficient (MFCC) for feature extraction and HMM for the stimation of parametes. They obtained an overall performance of 90.62%, 98.01% and 97.99% for sentence correction, word correction and word accuracy respectively.

Saini, Preeti, Parneet Kaur, and Mohit Dua [7] conducted experiment for Hindi using Hidden Markov Model Toolkit . The system recognized 113 isolated Hindi words collected for nine speakers. They obtained an overall accuracy of 96.61%

and for 10 state HMM gor 95.49%. Cini kurian and Kannan Balakrishnan has contributed to Malayalam speech recognition they developed a Malayalam digit speech recognition model which is speaker independent. The system employed Hidden Markov model (HMM) for recognition and Mel frequency cepstrum coefficient (MFCC) as feature for signal processing . Shyam.k Developed an automatic speech recognition system using CMUsphinx and its online API to to recognize malayalam commands and control Desktop.

CHAPTER 7

CONCLUSION

This project is to make a interaction module for humanoid robot in Malayalam. Interaction module will be capable of recognizing and responding to Malayalam queries. So communication with this module will be easy without language barrier. This work will be usefull for future works related to Malayalam speech. because this interaction module is not limited to humanoids only. We can use it in any application which needs communication module in Malayalam. As a future scope accuracy of the module can be increased by adding more data for training speech recognition system. Online learning capabilities can be used for making the module learn in real time and increase its accuracy. Current method do not have a mechanism to understand the meaning of the recognized speech. If the module have this capability the response phase will become more accurate.

Publications

1. *"Explaining and harnessing Adversarial Examples"* Ian J. Goodfellow, Jonathon Shlens Christian Szegedy, ICLR, 2015

2. *"Adversarial Reprogramming of Neural Networks* , Gamaleldin F. Elsayed, Ian Goodfellow, Jascha Sohl-Dickstein, ArXiv e-prints, 2018

3. *" Ensemble Adversarial Training: Attacks and Defenses"* , F. Tramr et al. , ArXiv e-prints, 2017

4. *"Towards Deep learning model resistant to adversarial attacks"* , Aleksander M, Ludwig Schmidt, ICLR, 2018