INDIAN INSTITUTE OF TECHNOLOGY, GUWAHATI

Department of Computer Science And Engineering

Project Report On

SPEECH BASED PLAYLIST

Based on Speech Recognition System

Submitted To:
Prof. (Dr.) Pradip K. Das

Submitted By:
Vijay Purohit (214101058)

For course fulfilment of CS 566: Speech Processing

Acknowledgement

This project is the submission for the course fulfilment requirement of the CS 566 - Speech Processing subject. We feel a deep sense of pleasure to acknowledge our gratitude to Prof. Pradip K. Das for providing us with an exciting project work experience in his subject. His timely guidance at every step throughout the project was encouraging and pushed our limits in completing our project. We would also like to thank the Teaching Assistants of the subject, who were always ready to clear our doubts. Finally, We are indebted to our classmates. Without their support, project completion within a strict deadline was not possible. As our unity motto, Apes Together Strong.

- Vijay Purohit (214101058)

Contents

List of Figures

1 ABSTRACT

Speech Based Playlist provides a speech-based solution to select a playlist of a particular language by speaking its name. The Application then displays five songs of the selected language. The user can again speak the index to select that song by speaking out the index name. It uses the concepts of the Hidden Markov Model to store the speech phenomenon and compare the new sample of the speech data with three seconds with the existing HMM models to detect the word spoken. The project can be further extended to support more songs in future. The Application is developed in C/C++ in Visual Studio IDE.

2 INTRODUCTON

2.1 What is Speech Recognition?

Speech recognition is an interdisciplinary sub-field of computer science and computational linguistics that develops methodologies and technologies that enable the recognition and translation of spoken language into text by computers. Some speech recognition systems require ”training” where an individual speaker reads text or isolated vocabulary into the system. Modern general-purpose speech recognition systems are based on Hidden Markov Models.

2.2 Our Project

This project uses Hidden Markov Model to detect the spoken word out of the vocabulary present. Voculabury in this Speech-based Project consists of Indices (zero, one, two, three, four) and languages (Hindi, English, Telegu). More languages can be further added to a fixed number, but the number of songs is currently fixed to five to show the developed prototype. Speaking the language will detect that word and show the playlist of that language. Then Speaking any one of the indexes will play that song number. The project includes facilities to train these words for new speakers. It also includes live training of the words.

2.3 Future Improvements

Future Improvements. The project is currently developed using C/C++ and is a bit system dependent. Future improvements include developing it into an independent system component, including more languages and the dynamic number of songs in the playlist, which are currently fixed as five. Also, the UI of the application can be made more dynamic and rich. Other multithreading supported High-level languages can be explored to run the training and testing components parallelly.

3 EXPERIMENTAL SETUP

Basic requirements for this project are as follows-

- Windows OS 10.
- Microsoft Visual Studio 2010.
- C++11 integrated with VS2010.
- Command Line Recording Module.
- A good Microphone

With the availability of above soft-wares, we further proceed in modelling the logic. The prerequisites of this project are:

- Basic i/o operations on File.
- Pre-processing of speech data. Generating Coefficients.
- Feature extraction.
- Modelling of extracted feature.
- Enhancing model.

4 PROPOSED TECHNIQUES

4.1 Flowchart

Figure 1 is a flowchart of the project. Flowchart can be referred for successful execution of the project.

4.2 Model description

We are using the famous Hidden Markov Model for speech recognition. Hidden Markov Model is a probabilistic model used to derive the probabilistic characteristic of any random process. We use Cepstral Coefficients to represent the speech properties. We take all such cepstral coefficients generated by preprocessing the speech frames and build a codebook that helps in generating the observation sequences. The codebook contains 30 speech samples for each word.
We start with a feed-forward model and use the word observation sequences one by one to converge the model to its optimal value. Later on, we average out all the converged models of that word that save it to a hard disk.
While testing, we score each model using the Forward Process and pick the word with the highest score as a resultant word. Stress is present since speech signals depend significantly on the environment; therefore, live testing might not be excellent. However, we might get significantly better accuracy if we train the model live and test it immediately.

4.3 Modules

Appropriate log files are generated for each operation in their respective folder which can be further referred for debugging purpose.

1. **Observations Sequence Module:** It Generate Observation Sequence for each word and their all utterances for every training and testing files present.

2. **Convergence Module:** It Converge All the Word Models one by one using the training observation sequences generated earlier. For the new word it generate observation sequences from its training files and then converge the model for that particular word only.

3. **Testing Module:** It take the observation sequences of testing files and detect which word they belong to. It contains offline testing and live testing of the word.

    - Offline Testing involves pre-recorded files of the words and then detecting which word is correctly identified.

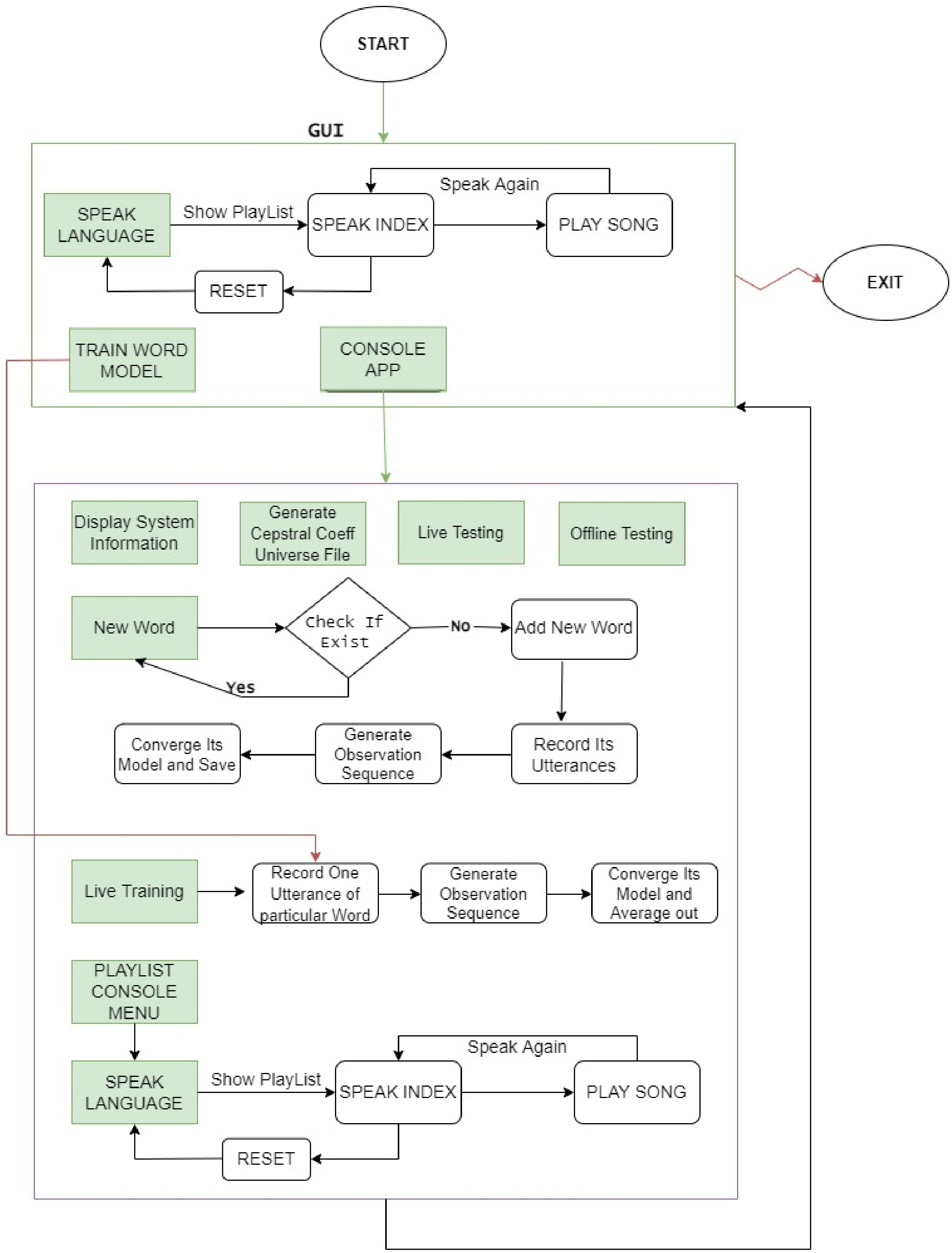


Figure 1: Flowchart of the Application

- Online Live Testing involves speaking one utterance using recording module and then checking it which word system is detecting.

4. **Training Module:** Here we can record the new utterances for the speaker which can be used for converging the module.

   - Live Training involves speaking one utterance of particular word then converging it by taking feed forward model. Later on averaging it with existing model of that word.
   - Recording Utterance Menu: where we can offline record each utterance of the all the words one by one. Later on it can be used for initial Convergence of the model.

5. **Playlist Module:** It is the Playlist menu which is first used to show playlist of the language and then play song from the songs listed for that language.

5 SNAPSHOTS

Figure 2: Playlist GUI: Languages

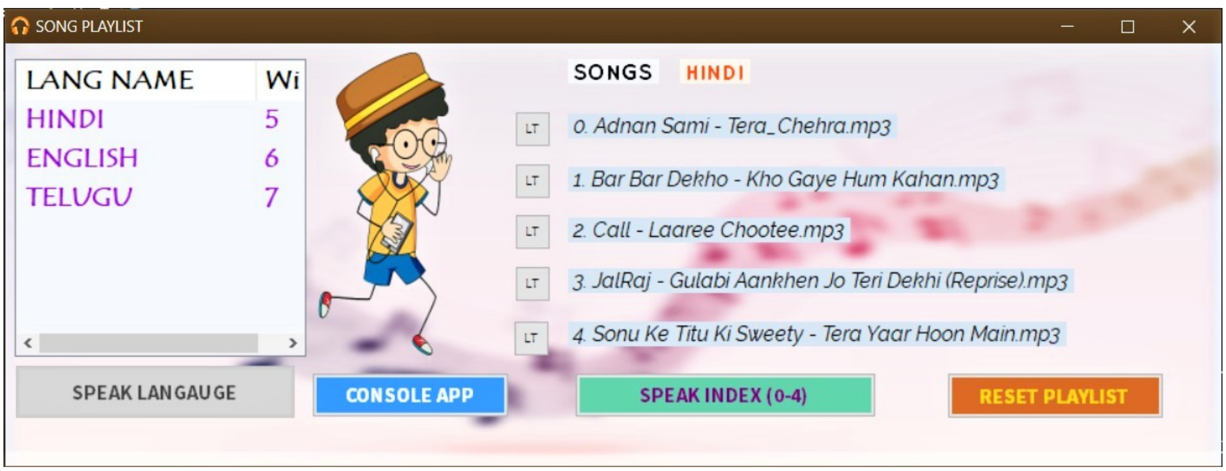


Figure 3: Playlist GUI: Hindi Playlist

Figure 4: Playlist GUI: Playing English Song

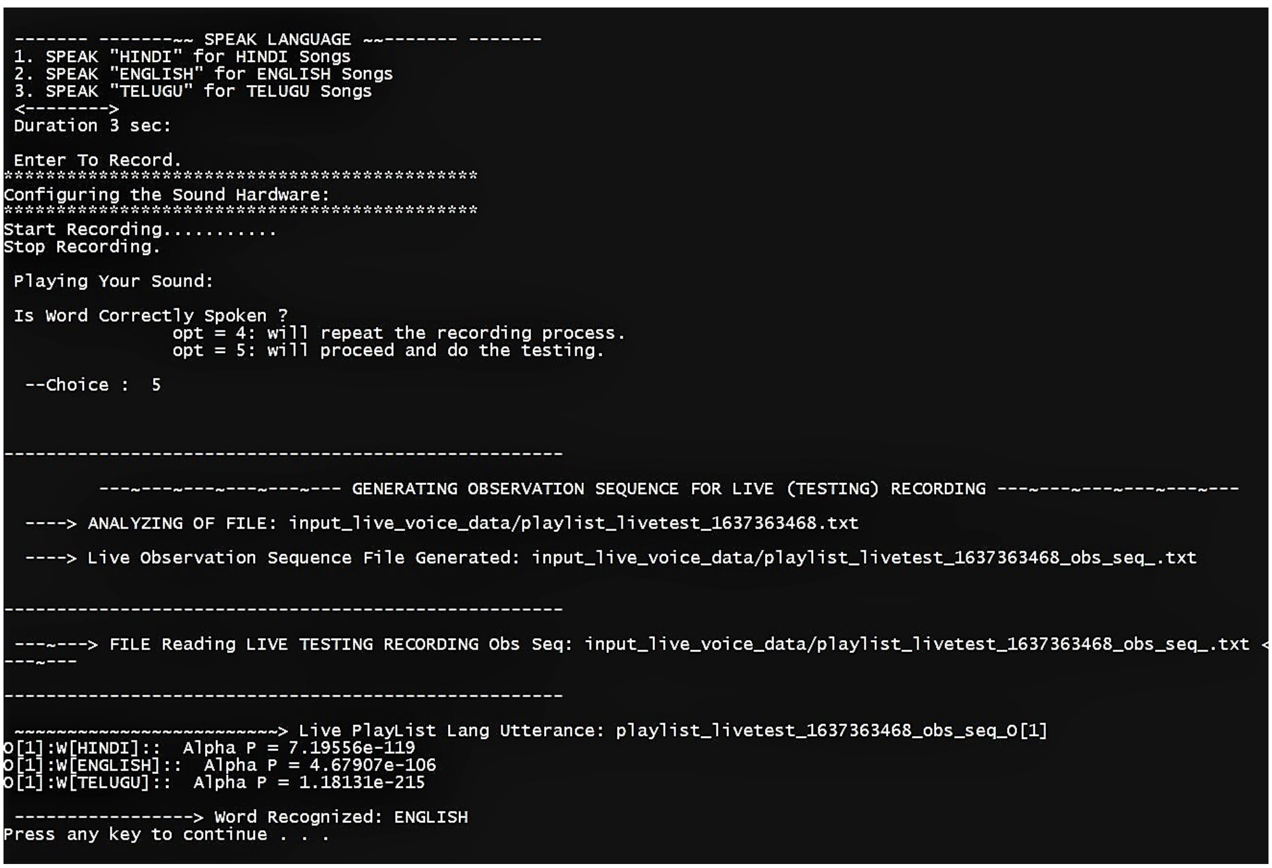


Figure 5: Playlist Console: Languages

```
------- -------~~ SPEAK INDEX ~~------- -------
"HINDI" SONGS::

0. [Adnan Sami - Tera_Chehra.mp3]

1. [Bar Bar Dekho - Kho Gaye Hum Kahan.mp3]

2. [Call - Laaree Chootee.mp3]

3. [JalRaj - Gulabi Aankhen Jo Teri Dekhi (Reprise).mp3]

4. [Sonu Ke Titu Ki Sweety - Tera Yaar Hoon Main.mp3]

<-------->
Duration 3 sec:

Enter To Record.
********************************************
Configuring the Sound Hardware:
********************************************
Start Recording...........
Stop Recording.

Playing Your Sound:

Is Word Correctly Spoken ?
               opt = 4: will repeat the recording process.
               opt = 5: will proceed and do the testing.

 --Choice :  5

----------------------------------------------------

        ---~----~----~----~--- GENERATING OBSERVATION SEQUENCE FOR LIVE (TESTING) RECORDING ---~----~----~----~---

 ----> ANALYZING OF FILE: input_live_voice_data/playlist_HINDI_livetest_1637363924.txt

 ----> Live Observation Sequence File Generated: input_live_voice_data/playlist_HINDI_livetest_1637363924_obs_seq_.txt


----------------------------------------------------

---~---> FILE Reading LIVE TESTING RECORDING Obs Seq: input_live_voice_data/playlist_HINDI_livetest_1637363924_obs_seq_
.txt <---~---

----------------------------------------------------

~~~~~~~~~~~~~~~~~~~~~~~~~> Live PlayList Lang Utterance: playlist_HINDI_livetest_1637363924_obs_seq_O[1]
O[1]:W[Zero]::  Alpha P = 6.97397e-112
O[1]:W[One]::  Alpha P = 3.85762e-126
O[1]:W[Two]::  Alpha P = 1.57448e-124
O[1]:W[Three]::  Alpha P = 0
O[1]:W[Four]::  Alpha P = 4.01813e-087

----------------> Word Recognized: Four
```

Figure 6: Playlist Console: Playing Hindi Song

```
<-------->****-------- WELCOME TO HMM --------****
-Common Settings are : -
P (=Q)(#of Cepstral Coefficients) : 12
Number of Words/Digits/HMM (totWords) : 8
Number of States per HMM (N) : 5
Number of Distinct Observation Symbols (M) or CodeBook Size (Y) : 32
Max Length of Observation Sequence (T) : 150
Number of Training Observations : 30
Number of Testing Observations : 20

Frame Size : 320
Tokhura Weights : 1.0(1) 3.0(2) 7.0(3) 13.0(4) 19.0(5) 22.0(6) 25.0(7) 33.0(8) 42.0(9) 50.0(10) 56.0(11) 61.0(12)
Amplitutde Value to Scale : 5000
Intital Header Lines Ignore Count : 5
Intital Samples to Ignore : 6400
Intital Noise Frames Count : 10
Noise to Energy Factor : 3
Sampling Rate of Recording: 16000

=>Total Words in HMM:: 8
-->W[0]:[Digit 0]
-->W[1]:[Digit 1]
-->W[2]:[Digit 2]
-->W[3]:[Digit 3]
-->W[4]:[Digit 4]
-->W[5]:[HINDI]
-->W[6]:[ENGLISH]
-->W[7]:[TELUGU]
----------------------------------------------------------------
```

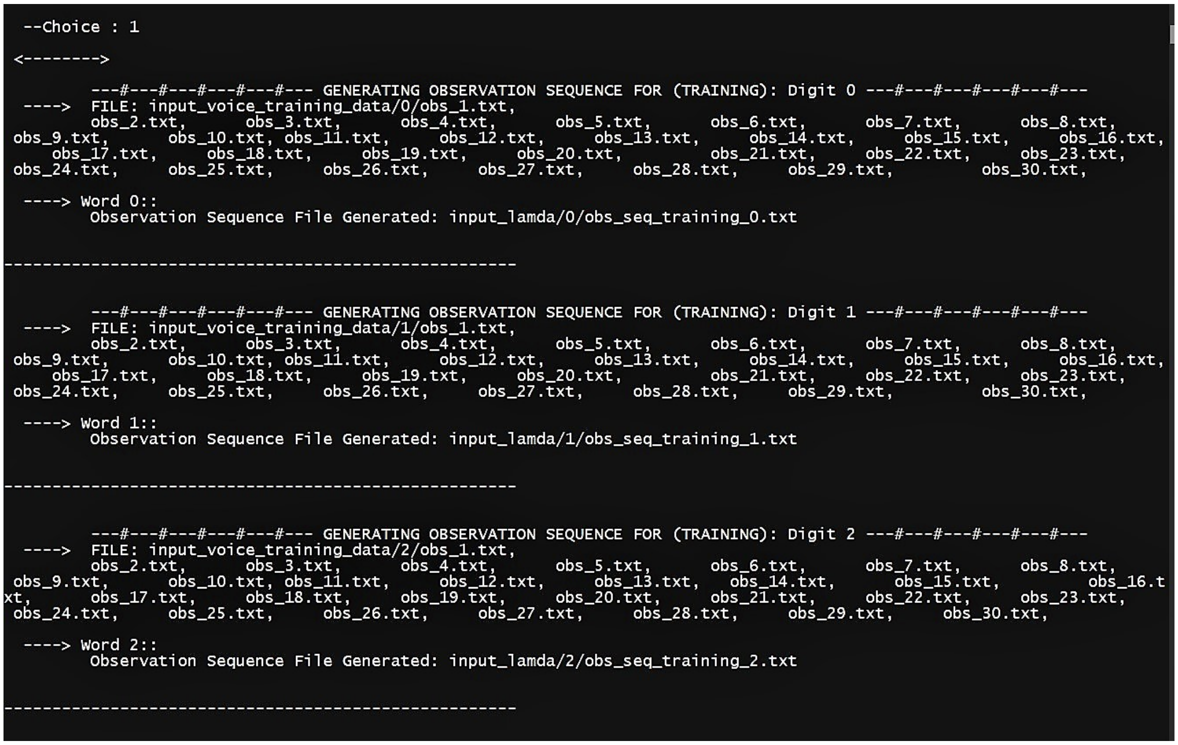
Figure 7: HMM: Common Settings Used

```
<-------->
=>Total Words in HMM:: 7
-->W[0]:[Digit 0]
-->W[1]:[Digit 1]
-->W[2]:[Digit 2]
-->W[3]:[Digit 3]
-->W[4]:[Digit 4]
-->W[5]:[HINDI]
-->W[6]:[ENGLISH]

Enter New Word To Add (Max Char: 101): Telugu

Word:Telugu

Word Directory Created: input_lamda/7/

        : input_live_voice_data/TRAINING/7/

        : input_live_voice_data/TESTING/7/

        : input_voice_training_data/7/

        : input_voice_testing_data/7/

        : output/Models/7/

        : SONGS/Telugu

=>Total Words in HMM:: 8
-->W[0]:[Digit 0]
-->W[1]:[Digit 1]
-->W[2]:[Digit 2]
-->W[3]:[Digit 3]
-->W[4]:[Digit 4]
-->W[5]:[HINDI]
-->W[6]:[ENGLISH]
-->W[7]:[Telugu]


NOTE: Please Make Sure to Add its Utterance in its Index Folder Manually or Use Record Utterance Menu

NOTE: Then Generate Observation Sequence and Converge the Model.
TRAINING Folder: input_voice_training_data/
TESTING Folder: input_voice_testing_data/
```

Figure 8: HMM: Add new word, Telegu

```
Total Iterations: 32    Alpha P = 4.34402e-061        Beta P = 4.34402e-061   Pstar P = 2.92243e-061
-###-###-###-###-###-###->>> Converged Model | Observation O[20] < ----------------------- -----------------------
Total Iterations: 9     Alpha P = 7.70039e-061        Beta P = 2.08008e-060   Pstar P = 4.96901e-062
-###-###-###-###-###-###->>> Converged Model | Observation O[21] < ----------------------- -----------------------
Total Iterations: 8     Alpha P = 1.4452e-063  Beta P = 5.47217e-063   Pstar P = 1.68898e-064
-###-###-###-###-###-###->>> Converged Model | Observation O[22] < ----------------------- -----------------------
Total Iterations: 6     Alpha P = 1.52644e-058        Beta P = 5.77479e-058   Pstar P = 1.55745e-060
-###-###-###-###-###-###->>> Converged Model | Observation O[23] < ----------------------- -----------------------
Total Iterations: 54    Alpha P = 1.06133e-059        Beta P = 2.06022e-059   Pstar P = 7.36358e-060
-###-###-###-###-###-###->>> Converged Model | Observation O[24] < ----------------------- -----------------------
Total Iterations: 107   Alpha P = 5.37943e-059        Beta P = 6.25472e-059   Pstar P = 1.20782e-059
-###-###-###-###-###-###->>> Converged Model | Observation O[25] < ----------------------- -----------------------
Total Iterations: 42    Alpha P = 1.04685e-056        Beta P = 2.82254e-056   Pstar P = 3.13081e-057
-###-###-###-###-###-###->>> Converged Model | Observation O[26] < ----------------------- -----------------------
Total Iterations: 64    Alpha P = 4.01696e-056        Beta P = 5.89192e-056   Pstar P = 1.8576e-056
-###-###-###-###-###-###->>> Converged Model | Observation O[27] < ----------------------- -----------------------
Total Iterations: 5     Alpha P = 4.07842e-058        Beta P = 1.02134e-057   Pstar P = 2.68239e-059
-###-###-###-###-###-###->>> Converged Model | Observation O[28] < ----------------------- -----------------------
Total Iterations: 146   Alpha P = 5.7427e-059  Beta P = 1.16004e-058   Pstar P = 7.14696e-060
-###-###-###-###-###-###->>> Converged Model | Observation O[29] < ----------------------- -----------------------
Total Iterations: 13    Alpha P = 2.49475e-056        Beta P = 3.60361e-056   Pstar P = 1.75256e-057
-###-###-###-###-###-###->>> Converged Model | Observation O[30] < ----------------------- -----------------------
Total Iterations: 52    Alpha P = 5.14511e-059        Beta P = 9.98756e-059   Pstar P = 3.5322e-059

        -------->> New Lambda Files Saved: output/Models/7/
        -------->> Convergence Done, Log File Generated: output/TELUGU_HMM_Converged_log.txt
```

Figure 9: HMM: Converge Model Output



```
 --Choice : 1

<-------->

        ---#---#---#---#---#--- GENERATING OBSERVATION SEQUENCE FOR (TRAINING): Digit 0 ---#---#---#---#---#---
 ---->  FILE: input_voice_training_data/0/obs_1.txt,
        obs_2.txt,      obs_3.txt,      obs_4.txt,      obs_5.txt,      obs_6.txt,      obs_7.txt,      obs_8.txt,
obs_9.txt,      obs_10.txt, obs_11.txt,     obs_12.txt,     obs_13.txt,     obs_14.txt,     obs_15.txt,     obs_16.txt,
    obs_17.txt,     obs_18.txt,     obs_19.txt,     obs_20.txt,         obs_21.txt,     obs_22.txt,     obs_23.txt,
obs_24.txt,     obs_25.txt,     obs_26.txt,     obs_27.txt,     obs_28.txt,     obs_29.txt,         obs_30.txt,

 ----> Word 0::
        Observation Sequence File Generated: input_lamda/0/obs_seq_training_0.txt


-----------------------------------------------------


        ---#---#---#---#---#--- GENERATING OBSERVATION SEQUENCE FOR (TRAINING): Digit 1 ---#---#---#---#---#---
 ---->  FILE: input_voice_training_data/1/obs_1.txt,
        obs_2.txt,      obs_3.txt,      obs_4.txt,      obs_5.txt,      obs_6.txt,      obs_7.txt,      obs_8.txt,
obs_9.txt,      obs_10.txt, obs_11.txt,     obs_12.txt,     obs_13.txt,     obs_14.txt,     obs_15.txt,     obs_16.txt,
    obs_17.txt,     obs_18.txt,     obs_19.txt,     obs_20.txt,     obs_21.txt,     obs_22.txt,     obs_23.txt,
obs_24.txt,     obs_25.txt,     obs_26.txt,     obs_27.txt,     obs_28.txt,     obs_29.txt,         obs_30.txt,

 ----> Word 1::
        Observation Sequence File Generated: input_lamda/1/obs_seq_training_1.txt


-----------------------------------------------------


        ---#---#---#---#---#--- GENERATING OBSERVATION SEQUENCE FOR (TRAINING): Digit 2 ---#---#---#---#---#---
 ---->  FILE: input_voice_training_data/2/obs_1.txt,
        obs_2.txt,      obs_3.txt,      obs_4.txt,      obs_5.txt,      obs_6.txt,      obs_7.txt,      obs_8.txt,
obs_9.txt,      obs_10.txt, obs_11.txt,     obs_12.txt,     obs_13.txt,   obs_14.txt,      obs_15.txt,         obs_16.t
xt,     obs_17.txt,     obs_18.txt,     obs_19.txt,     obs_20.txt,     obs_21.txt,     obs_22.txt,     obs_23.txt,
obs_24.txt,     obs_25.txt,     obs_26.txt,     obs_27.txt,     obs_28.txt,     obs_29.txt,     obs_30.txt,

 ----> Word 2::
        Observation Sequence File Generated: input_lamda/2/obs_seq_training_2.txt


-----------------------------------------------------
```

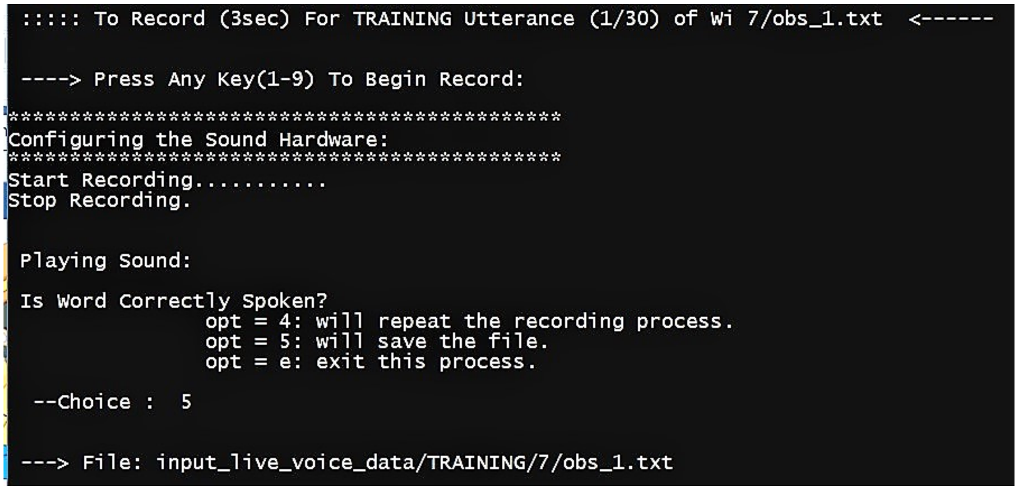
Figure 10: HMM: Observation Sequences Generation, Training

```
        ---#---#---#---#---#--- GENERATING OBSERVATION SEQUENCE FOR (TESTING): Digit 0 ---#---#---#---#---#---
  ---->  FILE: input_voice_testing_data/0/obs_1.txt,
         obs_2.txt,      obs_3.txt,      obs_4.txt,      obs_5.txt,      obs_6.txt,      obs_7.txt,      obs_8.txt,
obs_9.txt,      obs_10.txt,     obs_11.txt,     obs_12.txt,     obs_13.txt,     obs_14.txt,     obs_15.txt,     obs_16.
txt,     obs_17.txt,     obs_18.txt,     obs_19.txt,     obs_20.txt,

  ----> Word 0::
         Observation Sequence File Generated: input_lamda/0/obs_seq_testing_0.txt


-----------------------------------------------------


        ---#---#---#---#---#--- GENERATING OBSERVATION SEQUENCE FOR (TESTING): Digit 1 ---#---#---#---#---#---
  ---->  FILE: input_voice_testing_data/1/obs_1.txt,
         obs_2.txt,      obs_3.txt,      obs_4.txt,      obs_5.txt,      obs_6.txt,      obs_7.txt,      obs_8.txt,
obs_9.txt,      obs_10.txt,     obs_11.txt,     obs_12.txt,     obs_13.txt,     obs_14.txt,     obs_15.txt,     obs_16.
txt,     obs_17.txt,     obs_18.txt,     obs_19.txt,     obs_20.txt,

  ----> Word 1::
         Observation Sequence File Generated: input_lamda/1/obs_seq_testing_1.txt


-----------------------------------------------------


        ---#---#---#---#---#--- GENERATING OBSERVATION SEQUENCE FOR (TESTING): Digit 2 ---#---#---#---#---#---
  ---->  FILE: input_voice_testing_data/2/obs_1.txt,
         obs_2.txt,      obs_3.txt,      obs_4.txt,      obs_5.txt,      obs_6.txt,      obs_7.txt,      obs_8.txt,
obs_9.txt,      obs_10.txt,     obs_11.txt,     obs_12.txt,     obs_13.txt,     obs_14.txt,     obs_15.txt,     obs_16.
txt,     obs_17.txt,     obs_18.txt,     obs_19.txt,     obs_20.txt,

  ----> Word 2::
         Observation Sequence File Generated: input_lamda/2/obs_seq_testing_2.txt


-----------------------------------------------------


        ---#---#---#---#---#--- GENERATING OBSERVATION SEQUENCE FOR (TESTING): Digit 3 ---#---#---#---#---#---
  ---->  FILE: input_voice_testing_data/3/obs_1.txt,
         obs_2.txt,      obs_3.txt,      obs_4.txt,      obs_5.txt,      obs_6.txt,      obs_7.txt,      obs_8.txt,
obs_9.txt,      obs_10.txt,     obs_11.txt,     obs_12.txt,     obs_13.txt,     obs_14.txt,     obs_15.txt,     obs_16.
txt,     obs_17.txt,     obs_18.txt,     obs_19.txt,     obs_20.txt,

  ----> Word 3::
         Observation Sequence File Generated: input_lamda/3/obs_seq_testing_3.txt


-----------------------------------------------------
```

Figure 11: HMM: Observation Sequences Generation, Testing

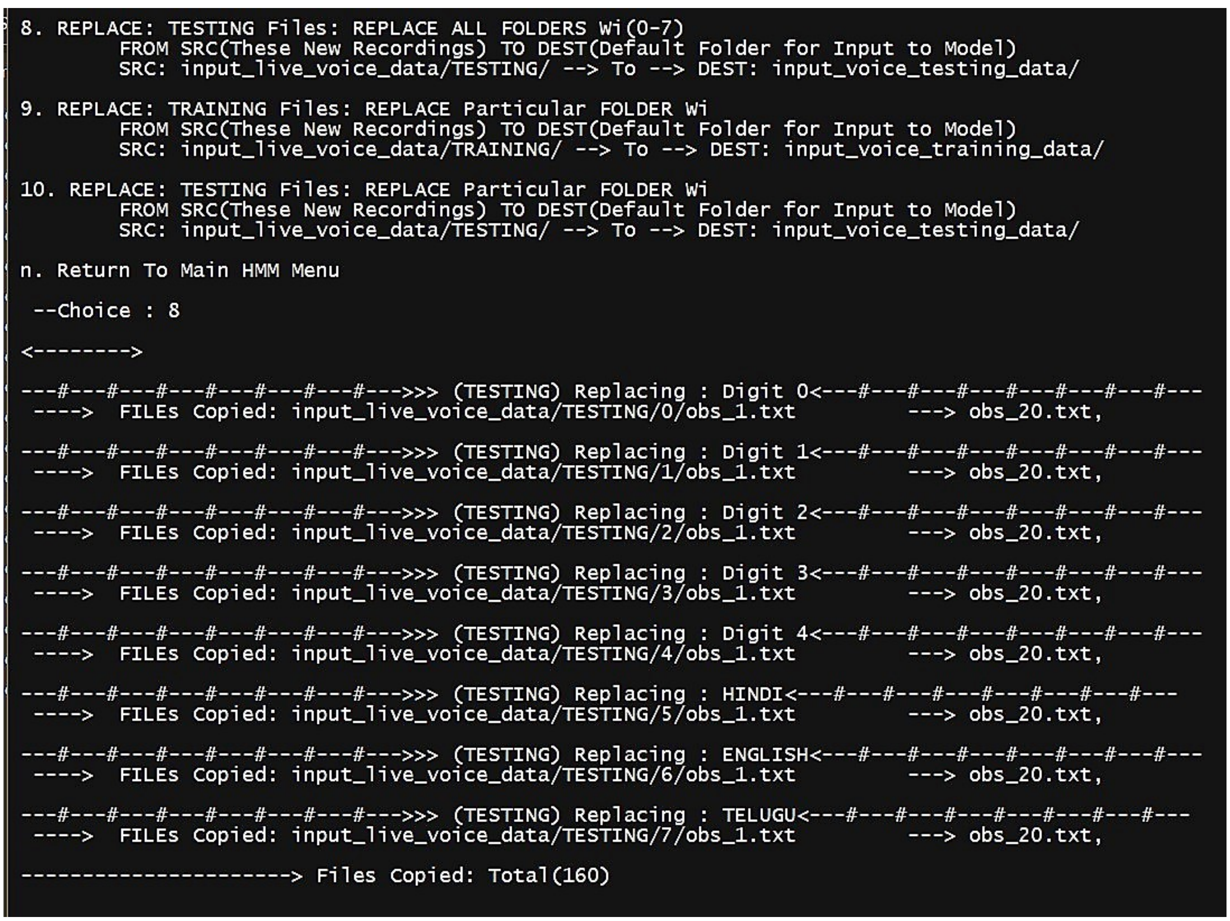
```
------- --------~~ Recording Utterances ~~-------- -------
1. RECORD: All DIGITS (0-7) Utterances #(30) for TRAINING Files. Total(240).
2. RECORD: All DIGITS (0-7) Utterances #(20) for TESTING Files. Total(160).

3. RECORD: Particular WORD (?) Utterances #(30) for TRAINING Files.
4. RECORD: Particular WORD (?) Utterances #(20) for TESTING Files.

5. RECORD: Particular WORD (?) Particular Utterance (?) for TRAINING Files. Total(1)
6. RECORD: Particular WORD (?) Particular Utterance (?) for TESTING Files. Total(1)

7. REPLACE: TRAINING Files: REPLACE ALL FOLDERS (0-7)
        FROM SRC(These New Recordings) TO DEST(Default Folder for Input to Model)
        SRC: input_live_voice_data/TRAINING/ --> To --> DEST: input_voice_training_data/
8. REPLACE: TESTING Files: REPLACE ALL FOLDERS (0-7)
        FROM SRC(These New Recordings) TO DEST(Default Folder for Input to Model)
        SRC: input_live_voice_data/TESTING/ --> To --> DEST: input_voice_testing_data/

9. REPLACE: TRAINING Files: REPLACE Particular FOLDER
        FROM SRC(These New Recordings) TO DEST(Default Folder for Input to Model)
        SRC: input_live_voice_data/TRAINING/ --> To --> DEST: input_voice_training_data/
10. REPLACE: TESTING Files: REPLACE Particular FOLDER
        FROM SRC(These New Recordings) TO DEST(Default Folder for Input to Model)
        SRC: input_live_voice_data/TESTING/ --> To --> DEST: input_voice_testing_data/

n. Return To Main HMM Menu

 --Choice :
```

Figure 12: HMM: Recording Utterances Menu



```
::::: To Record (3sec) For TRAINING Utterance (1/30) of Wi 7/obs_1.txt  <------

 ----> Press Any Key(1-9) To Begin Record:
*********************************************
Configuring the Sound Hardware:
*********************************************
Start Recording...........
Stop Recording.

 Playing Sound:

 Is Word Correctly Spoken?
                opt = 4: will repeat the recording process.
                opt = 5: will save the file.
                opt = e: exit this process.

  --Choice :  5

 ---> File: input_live_voice_data/TRAINING/7/obs_1.txt
```

Figure 13: HMM: Recording Utterances Menu, Recording one Utterance of word 7



```
8. REPLACE: TESTING Files: REPLACE ALL FOLDERS Wi(0-7)
        FROM SRC(These New Recordings) TO DEST(Default Folder for Input to Model)
        SRC: input_live_voice_data/TESTING/ --> To --> DEST: input_voice_testing_data/

9. REPLACE: TRAINING Files: REPLACE Particular FOLDER Wi
        FROM SRC(These New Recordings) TO DEST(Default Folder for Input to Model)
        SRC: input_live_voice_data/TRAINING/ --> To --> DEST: input_voice_training_data/

10. REPLACE: TESTING Files: REPLACE Particular FOLDER Wi
        FROM SRC(These New Recordings) TO DEST(Default Folder for Input to Model)
        SRC: input_live_voice_data/TESTING/ --> To --> DEST: input_voice_testing_data/

n. Return To Main HMM Menu

 --Choice : 8

<-------->

---#---#---#---#---#---#---#--->>> (TESTING) Replacing : Digit 0<---#---#---#---#---#---#---#---
 ---->  FILEs Copied: input_live_voice_data/TESTING/0/obs_1.txt         ---> obs_20.txt,

---#---#---#---#---#---#---#--->>> (TESTING) Replacing : Digit 1<---#---#---#---#---#---#---#---
 ---->  FILEs Copied: input_live_voice_data/TESTING/1/obs_1.txt         ---> obs_20.txt,

---#---#---#---#---#---#---#--->>> (TESTING) Replacing : Digit 2<---#---#---#---#---#---#---#---
 ---->  FILEs Copied: input_live_voice_data/TESTING/2/obs_1.txt         ---> obs_20.txt,

---#---#---#---#---#---#---#--->>> (TESTING) Replacing : Digit 3<---#---#---#---#---#---#---#---
 ---->  FILEs Copied: input_live_voice_data/TESTING/3/obs_1.txt         ---> obs_20.txt,

---#---#---#---#---#---#---#--->>> (TESTING) Replacing : Digit 4<---#---#---#---#---#---#---#---
 ---->  FILEs Copied: input_live_voice_data/TESTING/4/obs_1.txt         ---> obs_20.txt,

---#---#---#---#---#---#---#--->>> (TESTING) Replacing : HINDI<---#---#---#---#---#---#---#---
 ---->  FILEs Copied: input_live_voice_data/TESTING/5/obs_1.txt         ---> obs_20.txt,

---#---#---#---#---#---#---#--->>> (TESTING) Replacing : ENGLISH<---#---#---#---#---#---#---#---
 ---->  FILEs Copied: input_live_voice_data/TESTING/6/obs_1.txt         ---> obs_20.txt,

---#---#---#---#---#---#---#--->>> (TESTING) Replacing : TELUGU<---#---#---#---#---#---#---#---
 ---->  FILEs Copied: input_live_voice_data/TESTING/7/obs_1.txt         ---> obs_20.txt,

----------------------> Files Copied: Total(160)
```

Figure 14: HMM: Recording Utterances Menu, Moving Testing Utterances