Speech Recognition Based Automated Technical Support System

By

Adnan Ali

NUST201463904MSEECS60014F

Supervisor:

Dr. Muhammad Ali Tahir

School of Electrical Engineering and Computer Science

National University of Sciences and Technology

Islamabad, Pakistan

April 2018

Speech Recognition Based Automated Technical Support System

By

Adnan Ali

NUST201463904MSEECS60014F

Supervisor:

Dr. Muhammad Ali Tahir

Department of Computing

A thesis submitted in partial fulfillment of the requirement for the degree of Masters in Information Technology (MS IT)

In

School of Electrical Engineering and Computer Science

National University of Sciences and Technology (NUST)

Islamabad, Pakistan

April 2018

NUST School of Electrical Engineering and Computer Sciences

A center of excellence for quality education and research

Approval

Certified that the contents of thesis document titled " Speech Recognition Based Automated Technical Support System" submitted by Mr. Adnan Ali have been found satisfactory for the requirement of the degree.

Advisor: Dr. Muhammad Ali Tahir

Signature: _____

Date: _____

Committee Member1: Dr. Muhammad Muneeb ULLAH

Signature: _____

Date: _____

Committee Member2: Dr. Ahmad Salman

Signature: _____

Date: _____

Committee Member3: Dr. Muhammad Imran Malik

Signature: _____

Date: _____

1.1 Dedication

In the name of Allah the beneficent and merciful, on whom we all are dependent for eventual support and guidance. I would like to dedicate this thesis work to my supervisor **Dr. Muhammad Ali Tahir**, I was completely unaware of speech recognition field, he guided me, encouraged me that I can do this. Also to My Parents to giving me the opportunity so I can even read something and at last to myself.

CERTIFICATE OF ORIGINALITY

I hereby declare that the thesis titled " **Speech Recognition Based Automated Technical Support System**" is my own work and to the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any degree or diploma at NIIT or any other educational institute, except where due acknowledgment, is made in the thesis. Any contribution made to the research by others, with whom I have worked at NIIT, SEECS NUST or elsewhere, is explicitly acknowledged in the thesis.

I also declare that the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the project's design and conception or in style, presentation and linguistic is acknowledged. I also verified the originality of contents through plagiarism software.

Author Name: Adnan Ali

Signature: _____

1.2 Acknowledgment

First of all, I would pay my gratitude to **ALMIGHTY ALLAH** the most Merciful and the most Beneficent. He gave me health, strength, and passion to carry this research work. After this, I would like to pay my regards to my **parents** for their support and encouragement and prayers. I am sincerely grateful to my advisor **Dr. Muhammad Ali Tahir** for his continuous encouragement, guidance and precious time he dedicated to my research work. He has been a remarkable mentor and I am very thankful to him for his encouraging support, providing me the opportunity to work in one of the significant fields of today era, and helping me throughout the research work.

I am also thankful to my GEC members **Dr. Muhammad Muneeb ULLAH, Dr. Ahmad Salman** and **Dr. Muhammad Imran Malik** for being so kind and available whenever I needed their help.

Adnan Ali

Keywords:

ASR = Automatic Speech recognition

CLI = Command Line Interface

GUI = Graphical User Interface

HMM = Hidden Markov Model

LM = Language Model

MFCC = Mel-Frequency Cepstral Coefficients

WER = Word Error Rate

Table of Contents

List of Figures:

List of Tables

Abstract

Speech recognition is one of the significant topics in recent times. In Human computer interaction, we study the ways for efficient interaction between human and computer. Typing with a keyboard to touch screen, CLI to GUI, this interaction is getting better and better. The speech also plays an important role to make it more efficient not for just disable persons but also healthy persons. To interact with the computer in speech, the computer should be able to understand spoken words. For this purpose spoken data is converted into written data and then it is used for further processing. In this thesis, the primary focus in the Urdu language. Urdu data collected from multiple sources cleaned the data and then trained this data using CMUSphinx which is HMM base tool. More than 83 hours data by 181 speakers, is used for training and more than 8 hours data by 20 speakers is used for testing. Minimum achieved WER is 35.6% against 5 testing speakers and 44% for 20 speakers which is best in all published papers of Urdu base ASRs. After the training, the acoustic model is created which is used in two Android based application. First is, an automatic technical support system, where the user calls to get technical support from his/her internet provider. The system understands his question and replies with the appropriate answer. If the system is unable to understand the question it asks again to speak. The second application is command based application where the user gives the command to system and system understand and acts accordingly. Another module of this application is speech and type where the user speaks something and it is changed to written text then this text can be copied and shared. Another desktop application is also an Urdu base application it is a helper application to remove errors and to reduce the error rate. Where researcher can compare testing input text and recognized text

Chapter 1

1. Introduction of Speech Recognition

1.1 Introduction:

A human can communicate in different ways like speak, write and with body language. When it comes to human to human communication then all three ways are accepted and can be interpreted in different ways. But when it comes to communicating with machines then mostly used method is giving input by typing with the keyboard. The rise of machines gave the idea of communicating with machines via speech as a medium[1]. Automatic Speech Recognition (ASR) is a system where a person speaks to the computer, smartphone, smart screen or similar device and it recognizes his voice and converts it to text. As a result, one may have written text instead of typing with the keyboard. Automatic Speech recognition is stimulating and widespread area of research in Human computer Interaction. The purpose is to take voice as input via telephone or microphone and change it into the text as realistic as possible. This written text can be used for further processing and as input for further systems. Characteristics of a good ASR are, it should be independent of the speaker, speaking style, accent, grammar, syntax, dis-fluency and most important it should have large vocabulary up to all possible words of particular language [2]. Purpose of ASR is to convert an audio signal to text so it can be used as input for applications. To use speech recognition in different applications there are certain processes which need to be completed. Like creating an acoustic model of that specific language which you want to use in speech base application. To create acoustic model further require files are language model, dictionary (lexicon), phones, a written material called as transcription data and then recording of the transcription data.

1.2 Urdu:

There are total 6909 languages in the world and some are near extinction because only a few old speakers of these languages are alive [3]. With the death of them, language may also die. Between all the languages which are part of active research on speech, some languages with high numbers of speakers also don't have too much part in research, mostly because they belong to developing countries. Urdu is one of them. Urdu has more than 109 million speakers only in Pakistan and including other countries speakers, this value goes up to 160 million. It is the national language of Pakistan, also it is spoken in and understood in India, Bangladesh, Nepal, Mauritius and South Africa. In Pakistan, it is not only spoken but also have its use in education system until secondary school. Its writing style is as Arabic script, it contains Arabic, Persian and Sanskrit words [3]. In speaking Urdu is more often used as compared to writing. Second famous language for education system is English, which is used mostly in the higher education system. Pakistan has a literacy rate of 60% [4]. These are people who can speak and understand Urdu, In addition, there are also people who can understand and speak Urdu but

cannot read it. Developing Urdu speech dependent applications and tools may have a massive industry in Pakistan. This much population of speakers make it attention for research in this language. Some of the natural language based work are investigated for Urdu Language as, Urdu grammars Checker[5], Urdu text Classification[6], Urdu WordNet[7], Tokenization[8], word segmentation, Sentence Boundary detection[9], rule based stemmer[10][11], Morphology orientation[9], Named Entity Recognition[5], Part of speech tagging[12] and speech recognition. Here speech recognition is the main concern of this thesis.

1.3 Terminologies related to Speech:

To understand speech recognition in detail, there are some definition and terms which required to be understood. These terms are relevant to speech and used in speech base research.

1.3.1. Lexicon / Dictionary:

Lexicon is a file which contains all the words of that specific ASR system. All words which are present in transcription, it is necessary for speech recognition system to contain all words in the lexicon. Dictionary is a combination of words and their sounds, it contains the information that how a word is spoken. Words like آبادی and آتش are present in lexicon file along with their sounds. For instance word, آبادی is described as (A b A dd i) and آتش as (A td i S). One important thing is to remember, that all corpus words and their phonemes must be included in the dictionary. Even if a single word is not included, it will become error and training cannot be started. So all unique words of the corpus are included in the dictionary. It also assists as midway between language model and the acoustic model. Table contains some words and their phonemes.

Table 1.1: Example of Dictionary Words

Word	**Pronunciation**	**Word**	**Pronunciation**
موبلاءز	m o b I l A I z	داهنی	dd A h n i
موازنه	m U v A z n A	دارومدار	dd A r u m dd A r
جبریل	dZ y b y r i l	کھینچا	kh Y n tS A
کھیلوں	kh e l u~	کیلکولیشن	k Y l k U l e S y n

گالوں	g A l o~	نازیہ	n A z i A

1.3.2. Language Model:

It's a statistical model for the sequence of words. It contains word's combinations and their probabilities. Probabilities are counted from sample data and have flexibility. It is possible that few words can have any combination but it varies from one word to another word. Like "it is" and "is it". In this combination, one may have less probability and other may have high and vice versa. There is another edge of it. Suppose there are two numbers. Thirty four and twenty five. So there will be still chances of occurrence of thirty five. So statistical model is preferable than simple grammar [13]. Imagine someone is speaking and you are trying to guess what he/she will say further. "I am going to…" What could be next word? School, market, home, office or anything else. So it is helping for noisy environment, where, when the system cannot recognize any word it guesses it. This information is in the language model.

1.3.3. Phones & Phoneme:

A phone is any different speech sound or gesture, irrespective of whether the precise sound is critical to the meanings of words. Basic unit denoted by the acoustic model is a phone. For Example, آباءی is an Urdu word and combination of four different phones A, b, A and i. While, a phoneme is different from the phone it is a speech sound that, if it were exchanged with another phoneme, the meaning of the word will be changed. All used Phones in this thesis work are given below in Table 1.1.

Table 1.1: Phones Used in this Thesis Work

Phonemes	**Phonemes**	**Phonemes**	**Phonemes**	**Phonemes**	**Phonemes**	**Phonemes**	**Phonemes**	**Phonemes**	**Phonemes**
SIL	O	b	dd	gh	l	p	s	td	y
A	S	bh	ddh	h	m	ph	t	tdh	z
A~	U	d'	e	i	n	q	t'	u	

E	Y	d'h	e~	i~	nh	r	t'h	u~	
I	Y~	dZ	f	k	o	r'	tS	v	
N	Z	dZh	g	kh	o~	r'h	tSh	x	

1.3.4. Word Error Rate (WER):

This is a term used to present error rate. In recognition process, there could be three possibilities with a word. First is, it can be substituted with a new word. Like "I go to school" becomes "I go to stool" so in this sentence school is substituted with stool. This is called substitution. The second possibility is it can be deleted and not replaced with any other. Like "Email has been sent to your address" becomes "Email has sent to your address", in this sentence been is deleted. The third possibility is a new word is inserted, like "Snow looks beautiful but it can be dangerous to drive" is changed to "Snow looks beautiful and but it can be dangerous to drive". Here and is inserted. These all three types are errors who cause the WER to increase. The formula of it is.

$$WER = (S + D + I)/100 \qquad \text{Equation 1.1}$$

Here

- *S= Substitution*
- *D= Deletion*
- *I= Insertion*

If WER is 30%. It means 3 words are inserted, deleted or substituted. Another scenario can be that 2 substituted and 1 word is either deleted or inserted. In this way, it is possible to have WER more than 100%. In few of my simulations, I got more than 100% error rate.

1.3.5. Types of Speech utterances: [14]

Vocalization or speaking of the word is called utterance that denotes a single meaning to the computer. It can be one word, combination of many words, sentences, or multiple sentences. Here are types of it.

Isolated Words: This words recognition module typically require each statement to have silence on both sides of the window of the sample. This requires one utterance at a time, not only one word. This is fine when the user has to give one word at one time or also can be used

in commands. But it is not reasonable for multiple words inputs. It is easiest and simplest way because words boundaries are defined.

Connected Words: this is similar to isolated words but these are connected. It allows different utterances to perform together but small pause between these.

Continuous Speech: It is dictation. Where speakers speak continuously and give dictation to the computer. Words are spoken together without any pause. This is difficult type because for this type of ASR large amount of training data is required. As vocabulary and corpus grow, word sequence grows. Words are spoken together so it becomes difficult to find their boundaries. It requires special effort to find utterances boundaries. *My System is also based on this type of speech*.

Spontaneous Speech: It is closed to natural speaking and it can be difficult in terms that it may include stutters, false starts, non-words and also mispronunciations of words. This is not prepared the type of speech. The user speaks first and then later its transcription is written.

1.3.6. Types of Speaker Model: [14]

Every speaker can have different features. Including speaking style, accent, and pitch. Due to his/her gender, age, body structure, and personality. So when the system contains same person data in training and testing then there will be less error rate and accuracy will be better but when training and testing data is completely independent. Then WER will be comparatively high, here the assumption we make that spoken words are correct. There are two types of the model on the basis of the speaker.

Speaker dependent models: This system supposed to be for one person only where training and testing are from one person. It is generally more accurate and has less word error rate. If testing speakers got changed then error rate will increase.

Speaker independent models: In this type of model training and testing speakers are different. For example 10, 20 or any number of speakers used for training and 1 or any number of testing speakers, who are different from training speakers. This system is more flexible and flexible for new speakers. Generally, its error rate is larger than speaker dependent. [14]

1.4 Applications:

In order to talk with devices, we need better ASR, more accurate is better. ASR applications are increasing day by day from data entry, keyboard enhancement, command and control, searching audio documents, bio metrics, home automation, Google and its services like Google translate which support 110 languages [15],ok Google, spoken search queries, Google speech can be used for application development, Google assistant, Microsoft Windows is supporting it in latest versions of windows like Cortana a virtual assistant by Microsoft , Apple is using the name as Siri, Amazon is working on Alexa, there are lot of applications in android store

which are speech base. Furthermore text processing tools, text-to-speech, electronic dictionaries, computer-aided instructions and for a disabled person to the daily routine application, speech is taking over as a next-generation tool for communication with machines.

1.5 Hidden Markov Model: (HMM)

Speech has time-based structure and can be encoded as a sequence of spectral vectors bridging the audio frequency range, the hidden Markov model (HMM) provides a natural framework for building such models. The foundation of this goes to HMM and speech to 1970's. Working of the system is, Features are extracted from spoken words, and these are fixed size spoken vectors. Then on decoding step decoder tries to find those spoken words which were most expected generated words while speaking. Figure 1.1 shows the working of Hidden Markov Model base Speech recognition. While feature vectors are denoted with Y and recognized words with w.

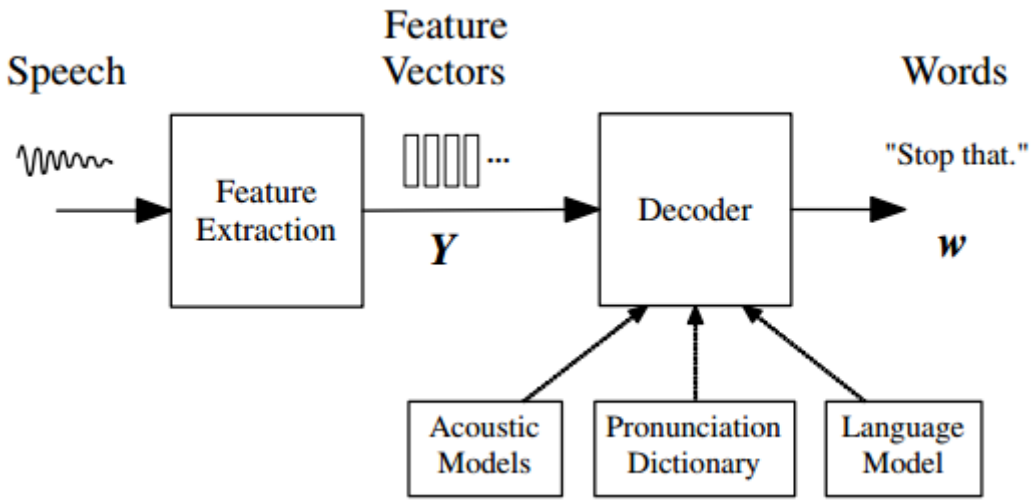


Figure 1.1: Working of Speech System

Each spoken word w is the composition of phones and this composition in a sequence is called pronunciation of that word. Many words can have same starting like

آءینی A i n i

آءینے A i n e

Starting of both words is same while at the end they are two different words.

To understand it with a simple example. Let's suppose there are three kinds of weather. Sunny rainy and foggy. One day there will be one weather and another day any other. In the middle of the day, weather doesn't change. Now we want to predict the weather for tomorrow. For

doing that we have to see history and observation. So let suppose, last three days weather was, sunny, sunny and rainy. Now, what are the chances that tomorrow will be rainy?

With Equation it can be denoted is.

$$P(w4 = rainy \mid w3 = foggy, w2 = sunny, w1 = sunny) \qquad \text{Equation 1.2}$$

Problem is when n value increases more statistics we need to predict. For n=5 then statistics should be 3^5=243. So Equation is

$$P(w_n | w_{n-1}, w_{n-2}, \ldots\ldots, w_1) \approx P(w_n \mid w_{n-1}) \qquad \text{Equation 1.3}$$

This is called a first-order Markov assumption. So tomorrow's weather will be at the base of today's weather. Probability of it, is given in Table 1.2

Table 1.2: Markov Model

		Tomorrow's Weather		
Today's Weather		Sunny	Rainy	Foggy
	Sunny	0.8	0.05	0.15
	Rainy	0.2	0.6	0.2
	Foggy	0.2	0.3	0.5

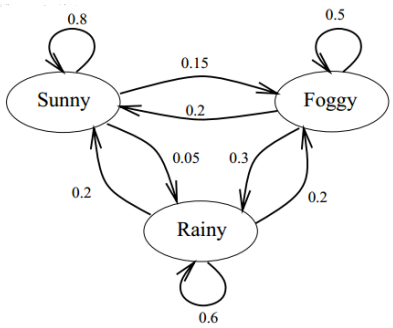


Figure 1.2: Markov Model

Now let's move to Hidden Markov Model. In Markov model, we have a sequence or chain of previous data. But now suppose you were locked in a room and don't know about the weather

now only hint you have is the person coming from outside is coming with an umbrella. Let's suppose the probability of umbrella in sunny weather 0.1, Rainy weather 0.8 and foggy 0.3. Now the actual weather is hidden from us so we use Bayes' rule:

$$P(w1, \dots, wn|\, u1, \dots., un) = \frac{P(u1,\dots,un|w2,\dots,wn)P(w2,\dots,wn)}{P(u1,\dots..un)} \qquad \text{Equation 1.4}$$

Now let's map it to speech recognition. The basic idea is to find a most likely string of words from recognized words. From above example, the Basic formula remains the same but the weather is replaced with words. In speech recognition, words are being predicted rather than weather. In the Language model example can be. Let's suppose someone says New York …. And there is one unrecognized word. That may be anything like news, city, newspaper, street or anything like this. So this is guessed by HMM that what probability of which word is. Hence most suitable word will be selected and recognized. A number of states in models. A model can have multiple states. Most of the times states are hidden. States are denoted with S={S1, S2, S3, ….. S_N}and State at Time t is Q_t. In weather examples Sunny, Rainy and foggy are states.

1.6 Neural Networks with ASR:

HMM is one of the classical methods in speech recognition, there is a lot of literature available, which used HMM for speech. But it suffers from problems in data fragmentation caused by decision trees[16]. Neural networks have been used along with HMM in previous years[17]. Neural Networks is a complex environment of simple computing elements which is capable of learning from examples. In recent times NN has been used individually in speech recognition. CMUSphinx doesn't use neural networks rather it use HMM. To make a mobile application required acoustic model is created with CMUSphinx that is the reason to use this tool and HMM base technique.

1.7 Conclusion:

This chapter gives the introduction of speech, its types, terms related to speech, lexicon, language model, phone, phonemes, word error rate, types of speaker model, Hidden Morkov model and Neural Network in ASR and application of speech in the current world.

Chapter 2

2. Literature Review

2.1 Introduction:

The core focus of this thesis work is Speech recognition in URDU, so in this chapter, it is explained that how much research got conducted in Urdu within recent years. Most research done by Pakistani researcher as Urdu is the national language of Pakistan. Some of the researchers from India have also contributed in Urdu, as in India it is also one of the famous languages.

2.2 Related Work:

There is not much work done in Urdu speech recognition due to lack of interest of researchers. Centre for Research in Urdu Language Processing (CRULP), Center for Language Engineering (CLE) [18] and National Language Authority [19] are major stakeholders in Urdu base applications. They have made Urdu keyboards for android and windows, Urdu games, Fonts. But not much work is done in Speech recognition. Notable speech works are "Recognizing Spoken Urdu Numbers Using Fourier Descriptor and Neural Networks with Matlab" this paper only consider Urdu counting numbers from zero (0) to nine (9) using tool Matlab and Simulink. This paper mainly focuses on feature extraction of these numbers [20]. There is another paper about speech corpus development, authors of this paper have done remarkable work by collecting data from 82 speakers, while participants are female and male both 40 and 42 respectively between ages 20 to 55 years. While recording is done in home and office environment [21].

This corpus is a combination of spontaneous speech and reading material. Recording environment was student labs, offices and for some speakers, it was home. They used a specific Dell laptop and software for recordings. From paper reading, it can be understood that was a special type of lab environment where table or chairs who may create creaking sounds were removed and pens which may create clicking sounds. While in case of office environment was normal but in labs case, labs were empty, which is not a normal scenario. Total recording data in hours is 44.5 hours 20.7 hours from females and 23.8 hours from males. This paper is the only collection of data, like our corpus this is also collected to train in CMUsphinx[13].

This paper [22] is an older version of above paper [21]. The difference is its data consist of only one speaker. Moreover, they describe that the Urdu language contains 62 phones and 250,047 possible tri-phones combination. From their analysis first they try to find tri-phone in words while considering that each word is followed by silence. While creating the corpus they considered to include 5681 words which were generated thorough algorithm. Then from these words, sentences were created by language experts of Urdu. They considered the difficulty level should be low for reading the sentence for a native Urdu speaker. Even some sentences which don't make a good sense was also included in the total of 725 sentences while each sentence contains minimum five words. The primary focus of the paper was to create phonetically rich corpus in order to include phones as much as possible [22].

An automatic speech recognition is made for Arabic language using CMUSphinx which is for continues speech. This system trained with 7 hours of data and have accuracy greater than 90% [23]. This paper's [24] authors are first who used Sphinx4 for Urdu. They used just 52 most common words and first ten digits. 10 speakers participated and each speaker recorded those words 10 times. So it makes 52x10x10=5200 utterances. After training then decoding the individual speaker's result the mean of WER was 5.33%. While lowest value was 0 and highest values was 10.0%. A speech-based system was developed on the basis of spontaneous speech. While Sphinx 3 was used to train and decode. But this data is comparatively small it is just 1 hour and 10 minutes data in 708 read base sentences which covers all the phones and tri-phones. Their data is grammatically correct but in some cases, it does not represent the basic structure of Urdu. In addition to this, there is also spontaneous speech corpus which is 1 hour and 59 minutes in form of interviews.

One important thing to note here is their language model is defined by actual training data. So when training data is changed then language model will be changed as well with it. The language model is created with SLM toolkit [25]. They tried different ratios between spontaneous and read data. So WER varies accordingly. Lowest achieved value is 18.8% when both were equal. While if read data is kept increasing then and spontaneous is decreasing the WER started gradually increasing and have a maximum value of 58.4%. But as I described their training data was small. Even combining both spontaneous and read base it is 2 hours and 37 minutes [26].

Development of language and acoustic model for the Urdu language is done in this paper where 81 are total speakers, 40 female and 41 male who participated in recordings, Combined data is about 45 hours data. While participants belong to Lahore and its suburb areas. The used tool is CMUSphinx [13] for training and decoding of data for speech recognition. To create language model SLM Toolkit [25] is used. Three different acoustic models were created, female only, male only and both male and female. Same speech corpus is used for creating the language model which made it to limited to less corpus. First, they created individual speaker acoustic model then combined all together. It is a good technique in order to get better results. So if the participant is causing too many errors, he/she might be skipped for good. Below is the table of this paper result set. Where we can see WER 60.2% to 79%. In refinement process adding diacritics make positive impact and error rate reduction. While there is no detail of testing data, whether it was 100 words or 1000 words. A total number of words and recognized word, in testing data, should have been presented in that paper to verify error rate. This value makes the WER. Which make the results unconvinced. [27]

Table 2.1 Results of the Paper [27]

	Voc. size	No. of training utterances	Tied states	LW	Word error rate
Best 41 M	12098	18835	5250	23	60.2
Worst 41 M	12098	18835	1000	23	64.9
Best 40 F	10981	11173	1000	17	65.6
Worst 40 F	10981	11173	1000	25	78.9
Best 81 M&F	14445	30983	5250	23	68.8
Worst 81 M&F	14445	30983	1000	23	79.0

Here is paper [28] from 2012 in which authors choose 250 remote words, including frequently using words and digits. These were narrow down to 19.3 million words to 5000 frequent then 250 most frequent words. Corpus collection is an important task in a sense what type of application you are going to create in terms of a number based application you may require numbers, for a health-based application you may need health based terminologies, for command and control application you may need a different type of words. But the problem is there is no such defined list of words. One approach is to include all 38 letters of Urdu in words, but like that, all letters should be once at the start, once in the center and once in end. Another way could have most frequently used words and make corpus from them. So they used later way for the paper. First, there were already conducted research by Center of Language Engineering [18] who identify 5000 words having a high frequency of 19.3 million Urdu words. Then they discarded prepositions but included digits, months of years and some more words because of significance in Urdu, their detail can be seen in the paper. But for recording, they used noise isolated environment which is not realistic. Mispronounced words were removed from recordings. 50 speakers from different age group were used for speaking all 250 words. While more on speaker 56% were female and remaining were male but half of them were native Urdu speakers and remaining half were not. So this paper just creates a corpus and record them there is no data training done in this paper.

There is another paper [29] in which authors made an application which deals continues speech and isolated words. But it is no very small data. Just 9 words and 5 sentences which were spoken by 8 speakers but 10 times. Total 540 utterances which are too small and then again these are divided into training and testing, 324 and 108 utterances respectively for each.

Training and testing data was mixed and chosen from same gathered data that is the reason for high accuracy 96.5%. If training and testing speakers were different then there must be less accuracy of the system, which they did not consider. Which make the system less realistic. Despite all these recording environments was close to realistic.

This paper [30] used data corpus of paper [21] and investigate and address the issues of paper [27]. Purpose of this paper is to create minimal balanced corpus instead of a large corpus. In multiple experiments, the author proves that if data is clean and balanced then better accuracy can be achieved. Yet It is not compulsory, that some phonemes accuracy can be better, even their data is little. But some phonemes despite having large data still may have less accuracy. A speech recognizer for four different languages of Pakistan Urdu, Punjabi, Sindhi and Pushto is developed in [31] it was the attempt when regional languages were included. It a single speaker data, one female 20 to 25 recorded the data 30 to 40 times for every single language. Door, light and fan turn on and off commands were recorded. The used tool is MATLAB, they made a kit to demonstrate results. LED on and off shows the results.

Extracting feature from the speech is done in [32]. Techniques are MFCC in full form Mel Frequency Cepstral Coefficients which attempt to simulate human ear reaction. Authors used three techniques to get features of speech. First technique is Support Vector Machine (SVM) which is criminative classification algorithm based on kernel, it gives the 73% accuracy and second is Random Forest which works on the basis of decision trees, it gives the accuracy of 63% and third but last is Linear Discriminant Analysis (LDA) which changes the data to discover a matrix and find to maximize the ratio between inter-class variance and the intra-class variance, results of this technique are the same RF. Discussing these techniques will be out of the scope of this paper. The authors of paper [33] used the corpus of [28] and recorded them from 8 male and 2 females. While males were non-native and female were native Urdu speakers. After recording file size 16 kb and 0.5 sec average recording time. For 100 words data set the achieved the data rate 16 to 26 while mean value is 21.8%. Total utterances are 1000 for 100 words. With 9 speakers training and 1 speaker testing error rate is 10%. While for the same case but 250 words with 9 and 1 ratio for training and the testing data rate is 22.3% and 25.34%.

The corpus of this paper [34] is about district names, days and time for travel domain purpose. So most particular value will be a destination so 44 major cities names of Pakistan were collected, the second value is the day, on which day a person wants to travel so names of days in English and Urdu. Then a number of seats to reserve. Either one two or three and so on so numbers were also included. Then information about time so commonly how time is spoken is included. Then further common words like yes no etc. More detail can be read in the paper. For recording the speech Interactive Voice Response (IVR) system was used with VOIP. They made a system with CentOS, Asterisk PBX and VOIP to collect data from the user. Age of speaker ranges 18 to 40, total 60 speakers 32 female and 28 males. This corpus further used to make bus reservation application which is published as a paper. Results of this paper [34] are 95.6% for laboratory results and while field results were 87.21%. Accent variation has been considered in [35] for district names of Pakistan. Their speech corpus consists more than nine

hours by 300 speakers from all over Pakistan to cover all six accents of Pakistan. Including Urdu, Punjabi, Pashto, Sindhi, Balochi, Seraiki. It is a good step to include recordings from different accents. All accents individual accuracy results are above 91.29% while average accuracy 92.87%. On field which include labs, classrooms, offices, campus parking, bus stand, cafeteria, and roads within campus accuracy is for accent dependent 60.06% and for accent independent 75.25%. The deployed system gave the accuracy of 71%.

Automatic language identification is topic of this paper [36] where authors describe that in India multiple languages are spoken so the task is to make such system which can identify the spoken language. There are multiple cues like phonology, morphology, syntax, and prosody on which a language can be separated from another language. While they used two cues phono-tactic and prosodic information with 72% and 68% accuracy. This is based on 7 Indian languages including Urdu.

This paper [37] is also based on multilingual speech recognition. The sequence to sequence model is being presented which can identify language without any external language requirement. They worked on 9 Indian languages while Urdu is one of them, with 196554 and 14486 training and testing utterances.

Some of Latest published papers are [38][39] and this [40] paper is about sentimental analysis in Urdu. Paper [38] is for speaker identification and they used Urdu data set for data training. While this paper [39] about language identification this work is in 6 languages and Urdu is one of them. This paper [41] is about speaker verification. Urdu is one of 5 languages used in this paper.

2.3 Conclusion:

This chapter contains research work published in years from 2008 to 2018. The main focus of this review is Urdu research. From small corpus development to large corpus development, limited words data training to continues data training, use of Urdu in sentimental analysis, speaker and language identification have been discussed here.

Chapter 3

3. Methodology and Tools

3.1 Introduction:

Acoustic model created here is from continues speech. There are a few steps which need to be followed in order to train an acoustic model. From tool selection to data collection, cleaning the data, recording the written data and recording environment each detail is discussed in this chapter. The used tool in this thesis work is CMUSphinx, there are published work about comparative study[42] of open source speech tools. Which states Kaldi as best open source tool and CMUSphinx as second best. Reason to choose CMUSphinx here is, its created acoustic model can be used in java based desktop applications, Sphinx4 is for this purpose, also its acoustic model can be used in mobile applications including Android and IOS. For mobile devices the tool is pocketsphinx. As I need to make a mobile application using acoustic model, for that purpose Sphinx is the best option.

3.2 Urdu Automatic Speech Recognition System:

Major components of an Urdu ASR are shown in Figure 3.1 and description is given below.

3.3.1 Front End:

Where user gives input with speaking something via microphone, it is Urdu so it is better, that spoken word, phrase or sentence should be in the Urdu language if it is in any other language it cannot be recognized correctly. Then features will be extracted from this speech signal. These features are in MFCC format and feature values are default as CMUSphinx.

3.3.2 Recognition:

Recognition step requires three main values, one is Language model, which help to identify next word and also help to identify that what is the probability of one word to be spoken after another word. Then there is dictionary/lexicon which contains all the words, if any word is not in lexicon it cannot be identified and acoustic model which is trained from the corpus.

3.3.3 Output:

After the recognition process, we have the output, which is written Urdu data. It is the result of input, it is not necessary that it is exactly as input, it can be different according to system accuracy. If system accuracy is good, like our system, then most of the words will be identified correctly. If accuracy is not good then chances are output will not be adjacent to input.

3.3.4 Decode:

Decoding is next step of output, in this process it is being checked that how many words got identified correctly. After this step, we have WER, word error rate. Higher the value lesser the system is accurate.

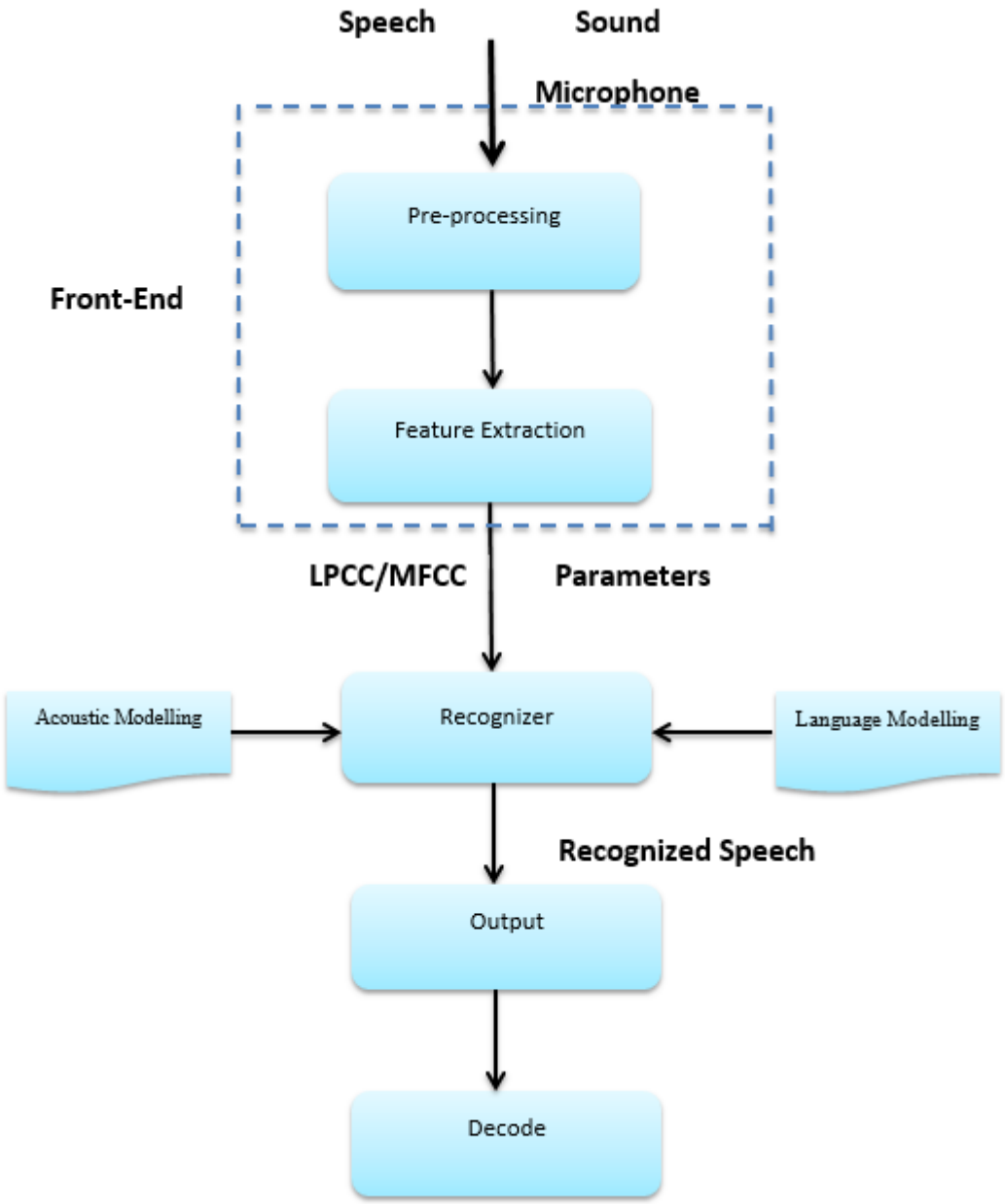


Figure 3.1: ASR System

3.4 CMUSphinx:

Sphinx4 is a tool made in java by Sphinx research group in Carnegie Mellon University, along with SUN, Mitsubishi and HP (Hewlett Packard). Further contributions are the University of California, Massachusetts Institute of technology. It is based on HMM. HMM is hidden Markov model which is a statistical model. Frontend, decoder and knowledge base are three

main module of sphinx4. The knowledge base is a further combination of three modules lexicon, an acoustic model, and language model.

3.4.1 Data Preparation:

In order to prepare the database for training in CMUSphinx here are required things. We need to have two folders name etc and wav.

While etc folder can be created from this command in Linux

sphinxtrain -t your_db setup

While "your_db" will be the name of the database you want to create. In my setup, I created multiple databases to test different aspects, but I will be using the example of "Urdu_training" database.

In my case values are given below and details are also defined.

```
├─ etc
│  ├─ urdudict.0.7a                    (Phonetic dictionary)
│  ├─ Urdu_training.phone              (Phoneset file)
│  ├─ Urdu_training.lm.bin             (Language model)
│  ├─ Urdu_training.filler             (List of fillers)
│  ├─ Urdu_training _train.fileids     (List of files for training)
│  ├─ Urdu_training _train.transcription (Transcription for training)
│  ├─ Urdu_training _test.fileids      (List of files for testing)
│  └─ Urdu_training.transcription      (Transcription for testing)
└─ wav
   ├─ speakeridentity                  (Folder)
   │   └─ 1a.wav                       (Recording of speech utterance)
   └─ speakeridentity                  (Folder)
       └─ 1b.wav                       (Recording of speech utterance)
```

3.4.2 Phonetic Dictionary (urdudict.0.7a):

This file has all words present in transcription file. It has one word per line and also its phonetic transcription. Sphinxtrain supports most of the special character like ":", "+" and "-" but don't support like "*", "/" [13]. It is better not to use a special character in the dictionary but only alphanumeric word are better to be placed in it. My dictionary contains 33708 Urdu words and their phonetic transcription in it. Numbers are converted into words. Like "اکسٹھ I k s y t'h", "چونسٹھ tS O n s y t'h" and "چون tS y v v y n".

3.4.3 Urdu_training.phone:

It contains the SIL for silence and one phone per line, these phones must be present in the dictionary [13]. If all phones are not in the dictionary, it will create error and training cannot be completed. My phone file has 56 phones for Urdu in it. The full list is shown in Table 1.2.

3.4.4 Language model file (Urdu_training.lm.bin):

Language model contains word combinations and their probabilities. Those probabilities are calculated from sample input data [13]. The language model is a file that contains the probability of a word after another word. Assume in this line "اس نے کھانا کھا لیا تھا" probability of کھا is higher than پیafter the word کھانا. A statistical model is recommended for continues data. Because this type of model is open for every word. Two language models were created. For table 4.1 there were 95784 entries in language model file. But for later tables' larger language model were used.

3.4.5 Filler Dictionary (Urdu_training.filler):

It contains the non-verbal sounds like hmm, um, noise or laughter etc. Our data don't contain such words like noise, laughter, hmm and um etc. but filler dictionary only contains phones for silence.

- <s> SIL
- </s> SIL
- <sil> SIL

3.4.6 Urdu_training_train.fileids:

This file contains the list of name of recordings or utterance IDs. There are total 29666 recording is my database for training, an example of first five lines is given below.

1. 14251556-01-20170407-ad/wav/1a
2. 14251556-01-20170407-ad/wav/2a
3. 14251556-01-20170407-ad/wav/3a
4. 14251556-01-20170407-ad/wav/4a
5. 14251556-01-20170407-ad/wav/5a

It is important that transcription and *fileids* file contains the same match. If it is not same then training will not happen and error will be shown.

3.4.7 Urdu_training_train.transcription:

This file contains a written form of spoken data, and also not speech words in exact order even non-speech sounds should be written in it. If it is not same as recording then this file will be rejected. For example, spoken words are 21 but written words are not equal to 21 then this file will become error and will not be included in the training. In this type of cases, there will be error and that specific line will not be included in the training. But in case of success, it will become the part of training.

Total lines in my recording database are 29666. Here are some example lines used in transcription file.

1. <s> تشویش کے باعث عملدرآمد نہیں کیا گیا ھے اس وقت واپڈا تربیلا کے آبی ذخیرے کے آپریشن کو اس طرح چلا رھا ھے کہ مٹی کے ڈیلٹا کی </s> (14251556-01-20170407-ad/wav/1a)

2. <s> ڈیم کی جانب بڑھنے کی رفتار کو کم کیا جاسکے بجلی پیدا کرنے کی ٹرباٰنوں کو محفوظ بنانے کے لیے سرنگ نمبر تین چار اور پانچ میں پانی </s> (14251556-01-20170407-ad/wav/2a)

3. <s> داخل ھونے کے مقامات کو بلند کیا جارھا ھے یہ تعمیراتی عمل اس وقت تکمیل کے مختلف مراحل میں ھے تربیلا ڈیم سے بالاٰی جانب دیا مربھاشا ڈیم </s> (14251556-01-20170407-ad/wav/3a)

4. <s> اور داسو ھاٰیڈرو پاور پروجیکٹ کی تعمیر سے بھی تربیلا کے آبی ذخیرے میں آنے والی مٹی کی مقدار میں کمی واقع ھوگی اور اس کی عمر میں </s> (14251556-01-20170407-ad/wav/4a)

5. <s> مزید پنتالیس سال کا اضافہ ھوجاٰے گا میں نے یہ معاملہ اس قدر وضاحت کے ساتھ اس لیے بیان کیا ھے کہ ایک درویش صفت باکمال شخصیت جناب </s> (14251556-01-20170407-ad/wav/5a)

There is actual transcription text between <s> </s> tags and path of wav file adjacent to this line is between the bracts ().

3.4.8 Urdu_training_test.fileids:

This file is same as Urdu_training_train.fileids but the difference here is this is used for testing. Here total files for testing are 2931. But different combinations of them are used for testing the data. Here are first five lines.

1. 16180856-01-20170407-ad/wav/A1
2. 16180856-01-20170407-ad/wav/A2
3. 16180856-01-20170407-ad/wav/A3
4. 16180856-01-20170407-ad/wav/A4
5. 16180856-01-20170407-ad/wav/A5

3.4.9 Urdu_training_test.transcription:

This file is same as Urdu_training_train.transcription but it is used for testing. Its lines are equal to Urdu_training_test.fileids.

First five adjacent to Urdu_training_train.fileids are given below.

1. <s> ایک مشترکہ بیان میں کہا ھے کہ ویڈیو سکینڈل منظر عام پر آنے سے موجودہ حکومت کے گڈ گورننس کے دعووں کی قلعی کھل گءی ھے ملزموں کی </s> (16180856-01-20170407-ad/wav/A1)

2. <s> طرف سے حوا کی بیٹیوں کے ساتھ انسانیت سوز سلوک نے عوام کے دلوں کو ھلا کر رکھ دیاھے اگر پھلی واردات پر ھی ملزموں کے خلاف کارواءی </s> (16180856-01-20170407-ad/wav/A2)

3. <s> عمل میں آ جاتی تو شاید حواء کی دوسری بیٹیوں کی عزتیں رسوا ھونے سے بچ جاتیں اور ملزمان اپنے انجام کو پھنچ جاتے انھوں نے وڈیو سکینڈل کے </s> (16180856-01-20170407-ad/wav/A3)

4. <s> تمام ملزمان ابھی تک گرفتار نہ ھونے پر پولیس کی نا اھلی قرار دیتے ھوءے کہا ھے کہ پھلے ھی غیر شادی شدہ لڑکیوں کے والدین اپنی لڑکیوں </s> (16180856-01-20170407-ad/wav/A4)

5. <s> کے مناسب رشتوں کے لیے پریشان تھے لیکن وڈیو سکینڈل نے والدین کی پریشانیوں کے ساتھ ساتھ انکے دلوں میں دھشت و خوف ڈیرے ڈال دیے ھیں انھوں </s> (16180856-01-20170407-ad/wav/A5)

3.4.10 Speech recordings (*.wav files):

Audio files are those files which are recordings of transcription files, these are required for both training and testing. File format should be wav and for the continues data, length should be from 5 to 30 seconds. Long recordings make recognition harder. It is compulsory that written and spoken data should be same. If it is not same then accuracy will decrease. Silence at start and end should not exceed 200ms. Along with file format, there are other things you need to consider. Sample rate 16 bit, 16 kHz [13] and mono is used for recordings as it is a requirement of CMUSphinx.

3.5 Data Collection for Recording:

The purpose was to create a new acoustic model for Urdu as there is no that big model is available right now. In this regards as per the direction of CMUSphinx documentation 50 hours of data from 200 different speakers is required to create for many speakers dictation [13]. More speaker leads to more diversity and system is more accurate. Because of different speaking style, age, education and gender matters in speech. In term of education level, different people can speak the same word in different style and while in term of gender male and female pitch and frequency are different for same age male and female. Between male and female speakers pitch is an essential factor. Pitch is determined by vocal fold's vibration. If the frequency is 300 Hz pitch it means vocal folds are vibrating 300 times in one second. In the sense of age and gender before adulthood of the male, the pitch is around 250 Hz. But when the puberty hits its values changes and becomes around 60 to 120 Hz. In female cases, its value is between 120 to 200Hz [43].

3.5.1 Collecting written corpus:

A large amount of text was required in order to complete 50 hours of data. In my case, it is more than 83 hours Collection of Urdu corpus. After collecting data, my observation and analysis are for recording one hour of data we need around 10000 to 11000 words to be spoken still it depends on speaker's speaking speed because everyone reads the text in his/her own way. Data collection is a really hard problem because of various reasons. Mostly electronically available data is either in gif and jpeg format [44] [45] some data are even in pdf format which is again a combination of graphics files. Even I found some pdf files which can be copied but when these are pasted text editors it becomes unknown Unicode characters. So, Written corpus was taken from Urdu blogs and news websites[46] [47] [48] [49] [50][51]. Taken data was in form of news, blogs, editorials, and articles published in these newspapers. Some of the data is collected from a social website (Facebook). This data was in thousands of lines and needed to be refined and cleaned in order to make the input of CMUSphinx.

3.5.2 Data Cleaning:

Collected data has a lot of mistakes or irrelevant information. It contained special characters like double and single quotes, commas, question marks sign of exclamation, asterisk, at the rate sign, starting and closing brackets, hyphen, semicolons, commas, backslash, forward slash, colon, starting and ending single, double quotes and other characters like this which are not the part of verbal speech and cannot be recorded and trained. e.g. "“,*@. All these characters were replaced with space or removed from the corpus. These characters are presented in Table 3.1

Table 3.1 Common Character Causes Errors

~	**`**	**!**	**@**	**#**
$	%	^	&	*
(	)	_	+	=
-	,	.	<	>
?	/	‘	ِ	-
ً	؟	؛	:	'
"	ّ	-	ٍ	ً

3.5.3 Converting it to CMUSphinx syntax:

CMUSphinx uses a specific format for training the data. So after cleaning the data next step was to change data into CMUSphinx format lines because before it was in paragraphs. After changing it into lines it becomes as each line contains 25 to 30 words.

Syntax of CMUSphinx is as

<s> Data to record and read as it will be used as input </s> (PATH of wav file)

Example:

<s> ایک مشترکہ بیان میں کہا ہے کہ ویڈیو سکینڈل منظر عام پر آنے سے موجودہ حکومت کے گڈ گورننس کے دعووں کی قلعی کھل گءی ہے ملزموں کی </s> (20170407-01-16180856-ad/wav/A1)

3.5.4 Recording and its Environment:

For recording the speaker's voice and accent, a tool used was "Audacity" [52]. It is open source audio software. This was for Computer using participants. Some participants used mobile (Android and Apple) default recording software for this purpose. Choosing the device was participant dependent. There was no special lab environment for recording as the purpose was to get more real-time data. No special instructions were given to participants expect to avoid

reading mistakes as much as possible. The noise factor was also normal just speaker voice should not be replaced with noise. All recordings were recorded with different frequency depending on ease of recorders. Then changed to 16 kHz and mono channel as required by CMUSphinx [13].

3.5.5 Language Model Creation:

SRI Language Modeling Toolkit SRILM [25] is used to make language model. It is a tool to create statistical base language model for speech and other purposes. It is based on N-gram statistics, it considered respectively ngram-count and ngram. A vocabulary file containing 33000+ words was also given in command so words added in LM will be matched from the vocabulary. The input data used to create this LM was different from training and testing data. It was also taken from multiple sources.

3.6 Conclusion:

This chapter explains CMUSphinx tool and data creation for this tool. At the end of this chapter, data is ready for simulations and trainings. Written data collection then its recording and cleaning the data. On another side creating the language model from sample data. All details have been discussed in this chapter.

Chapter 4

4 Simulations and Results

4.1 Introduction:

Until completion of the previous chapter, data is ready for simulations, in this chapter simulations and results are discussed. Specification of the system used for training and decoding are also given in detail in this chapter. Multiple simulations were run on multiple computer nodes of supercomputer to get the best accuracy.

Table 4.1 Vocabulary and Senones/Densities values

Vocabulary	**Audio in database / hours**	**Senones**	**Densities**
20	5	200	8
100	20	2000	8
5000	30	4000	16
20000	80	4000	32
60000	200	6000	16
60000	2000	12000	64

Table 4.1 is taken from official website [13] of CMUSphinx which shows, Which type of values should be for how much amount of data. So I tried with different combinations of these values to get better results.

4.2 Training and Decoding System:

For Table 4.2 used computer is Haier y11b running windows 8.1 as a host operating system and Ubuntu 16.10 as a guest operating system in the virtual box. Specifications of this computer are

- CPU: Intel Core 5Y10C
- RAM (G): 4
- SSD: Yes
- SSD capacity (G): 32
- HDD capacity (G): 500

For remaining table's training, testing and decoding used system is supercomputer located in NUST. Specification of supercomputer for each end node

- Two 2.27 GHz 64bit Intel 4-core Xeon E5520 processors
- 8 physical cores (16 logical cores if using Hyper-Threading)
- 24GB DDR3 RAM
- 2 x 250GB SATA Hard Drives
- CentOS 6.5 Operating System

While 10 threads were used for data training and decoding. Still, it took average more than 8 to 9 hours to complete one training with 83-hour data. While for decoding average time was 6 to 7 hours depending on testing data.

In below these scenarios I am changing "training and testing" data. Training data is increased in gradually. While testing data is increasing as well as replaced with newer data to test different result combination.

4.3 Training while Training Data increasing.

4.3.1 Scenario One

Training Data:

Training data is collected from 55 Speakers which is 23 hours and 42 minutes. Data corpus is generated from, stories, articles and news website as mentioned in chapter 3.

Testing Data:

Testing data is collected from 1 Speaker which is 0 hours and 29.38 minutes. This data is consisting of a novel base story. This data is completely different from training data.

Language Model:

Language model used for this scenario is small. It consists of 9200 lines. While each line contains average 30 words and total 95784 entries in language model file.

Senones:

To compose detector for tri-phones, short sound detectors are used which are called senones. CMUSphinx uses usually 4000 senones for tri-phones.

Densities used are 8, 16 and 32. For large to small vocabulary, its value will be increasing. For small data, the value should be 8, for larger 16, 32 or 64. It can be any power of 2.

Table 4.2: 55 Speakers Training Data

Sr#	Transcription Data (Hours)	No of Speakers	Testing Data (Minutes	No of Speakers	Senones	Densities	Word Error Rate
1	23.7	55	29.38	1	1000	8	94% (2990/3181)
2	23.7	55	29.38	1	2000	8	95.7% (3043/3181)
3	23.7	55	29.38	1	3000	8	99.7% (3171/3181)
4	23.7	55	29.38	1	4000	8	99.1% (3151/3181)
5	**23.7**	**55**	**29.38**	**1**	**1000**	**16**	**93.3% (2967/3181)**
6	23.7	55	29.38	1	2000	16	96.6% (3071/3181)
7	23.7	55	29.38	1	3000	16	97.6% (3104/3181)
8	23.7	55	29.38	1	4000	16	100% (3181/3181)
9	**23.7**	**55**	**29.38**	**1**	**1000**	**32**	**92.5% (2942/3181)**
10	23.7	55	29.38	1	2000	32	96.5% (3070/3181)
11	23.7	55	29.38	1	3000	32	99.2% (3155/3181)
12	23.7	55	29.38	1	4000	32	100.8% (3204/3181)

Description:

Data is trained with different combinations of senones and densities. As shown in Table 4.2. Three different values of densities are used in this table, values are 8, 16 and 32.While 4 different type of senones are used which are 1000, 2000, 3000 and 4000. Both combinations make 12 different rows of the table. Same training and testing data are used but here we can observe WER 92.5% minimum to 100.8% maximum. So, we can see changing densities and senones values are affecting simulation results.

The best result we are getting is 92.5% with 1000 senones and 32 densities. While 2nd best is 93.3% which are 1000 senones and 16 densities.

4.3.2 Scenario Two

Training Data:

Training data is collected from 105 Speakers which is 48 hours and 53 minutes. This data corpus is an addition to scenario one training data again this addition is taken from news websites, articles, stories etc. Recording participants are from Punjab, Pakistan. Each training person has 25 to 35 minutes recording contribution.

Testing Data:

Testing data is collected from 5 Speakers which is 0 hours and 29.38 minutes for Table 4.3. While for Table 4.3 speakers and testing corpus are completely different from Table 4.3. Testing time is also not same which 44.89 minutes here is. These minutes are approximately equally divided between these speakers.

Table 4.3: 105 Speakers Training Data

Sr#	Transcription Data (Hours)	No of Speakers	Testing Data (Minutes)	No of Speakers	Senones	Densities	Word Error Rate
1	48.89	105	46.16	5	3000	8	42.6% (3007/7064)
2	48.89	105	46.16	5	1000	8	46.7% (3300/7064)
3	48.89	105	46.16	5	4000	16	43.1% (3045/7064)
4	**48.89**	**105**	**46.16**	**5**	**3000**	**16**	**42.0% (2966/7064)**
5	48.89	105	46.16	5	2000	16	42.4% (2997/7064)
6	48.89	105	46.16	5	1000	16	42.3% (2988/7064)
7	48.89	105	46.16	5	4000	32	43.8% (3091/7064)

8	48.89	105	46.16	5	3000	32	42.5% (3000/7064)
9	**48.89**	**105**	**46.16**	**5**	**2000**	**32**	**42.1% (2976/7064)**
10	48.89	105	46.16	5	1000	32	42.7% (3017/7064

Description:

Now it is 48 hours and 53 minutes of training. For testing, 5 different speakers are chosen who participated approximately equal time in testing data. As we can see in Table 4.3 WER is changed to 42%. Best results in this scenario are 42%. Which is 50% less than Table 4.2.

So increasing training data from (23 hours to 48 hours) and changing testing speakers (1 to 5), gave positive results. Another thing we can observe from these two tables is, now best results (42%) have different values of senones and densities. 3000 and 16 respectively. While in Table 4.2 these were 1000 and 32.

Table 4.4: 105 Speakers Training Data with Different Testing Data

Sr#	Transcription Data (Hours)	No of Speakers	Testing Data (Minutes	No of Speakers	Senones	Densities	Word Error Rate
1	48.89	105	44.89	5	3000	8	41.1% (2887/7018)
2	48.89	105	44.89	5	1000	8	47.0% (3294/7018)
3	48.89	105	44.89	5	4000	16	41.7% (2923/7018)
4	48.89	105	44.89	5	3000	16	41.3% (2901/7018)
5	48.89	105	44.89	5	2000	16	41.2% (2889/7018)
6	48.89	105	44.89	5	1000	16	41.2% (2892/7018)
7	48.89	105	44.89	5	4000	32	41.5% (2913/7018)

8	48.89	105	44.89	5	3000	32	40.6% (2847/7018)
9	**48.89**	**105**	**44.89**	**5**	**2000**	**32**	**39.8% (2795/7018)**
10	48.89	105	44.89	5	1000	32	40.6% (2852/7018)

4.3.2.1 Description:

Table 4.4 contains same training environment as Table 4.3. But testing data is different from Table 4.3. In this table, we can observe the lowest value of WER, which is 39.8% against the senones value 2000 and densities value 32. Second, best is 40.6% which from two different simulations results. Senones and Densities are respectively 3000, 32 and 100, 32. Another thing can be observed from these results if we change testing data, the error rate will change either in the positive or negative way.

4.3.2.2 Comparison of Table 4.3 and Table 4.4:

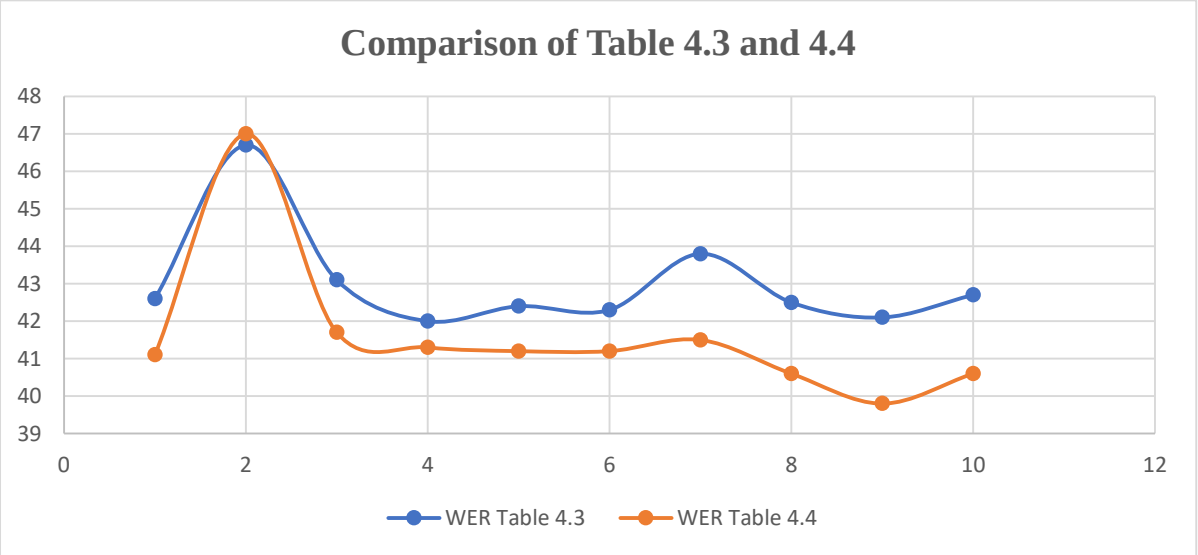


Figure 4.1: Table 4.3 and Table 4.4 Comparison

4.3.3 Scenario Three

Training Data Corpus:

In addition to scenario one and scenario two data, these data are data of 76 speaker's recordings. Which consisting of 25 to 30 minutes of recording and average 175 lines while each line contains 28 to 30 words.

Scenario one and two speakers = 105

Scenario three speakers = 75

Total = 181

As total we have 83 hours and 7.8 minutes with 181 speakers.

Testing Data Corpus:

To compare the results of small training data and large training data, I used same data corpus here as it was used in Table 4.4

Table 4.5:181 Speakers Training Data

Sr#	Transcription Data (Hours)	No of Speakers	Testing Data (Minutes)	No of Speakers	Senones	Densities	Word Error Rate
1	83.13 h	181	44.89	5	3000	8	41.6% (2920/7018)
2	83.13 h	181	44.89	5	1000	8	44.3% (3108/7018)
3	83.13 h	181	44.89	5	2000	16	38.1% (2673/7018)
4	83.13 h	181	44.89	5	4000	16	37.5% (2631/7018)
5	83.13 h	181	44.89	5	3000	16	37.8% (2650/7018)
6	83.13 h	181	44.89	5	1000	16	39.9% (2798/7018)
7	**83.13 h**	**181**	**44.89**	**5**	**4000**	**32**	**36.6% (2569/7018)**
8	**83.13 h**	**181**	**44.89**	**5**	**3000**	**32**	**36.9% (2590/7018)**

9	**83.13 h**	**181**	**44.89**	**5**	**2000**	**32**	**35.7% (2504/7018)**
10	83.13 h	181	44.89	5	1000	32	38.7% (2716/7018)

Description:

In this Table 4.5 simulations, training data is increased to 83 hours with total 181 speakers. But when I tested same testing data which was in

Table 4.4. It can be clearly seen in the table we have less error rate with senones and densities of 2000 and 32 and it is now 35.7% which is best so far in all simulation done till yet. Closest of it is 36.6% if senones are 3000.

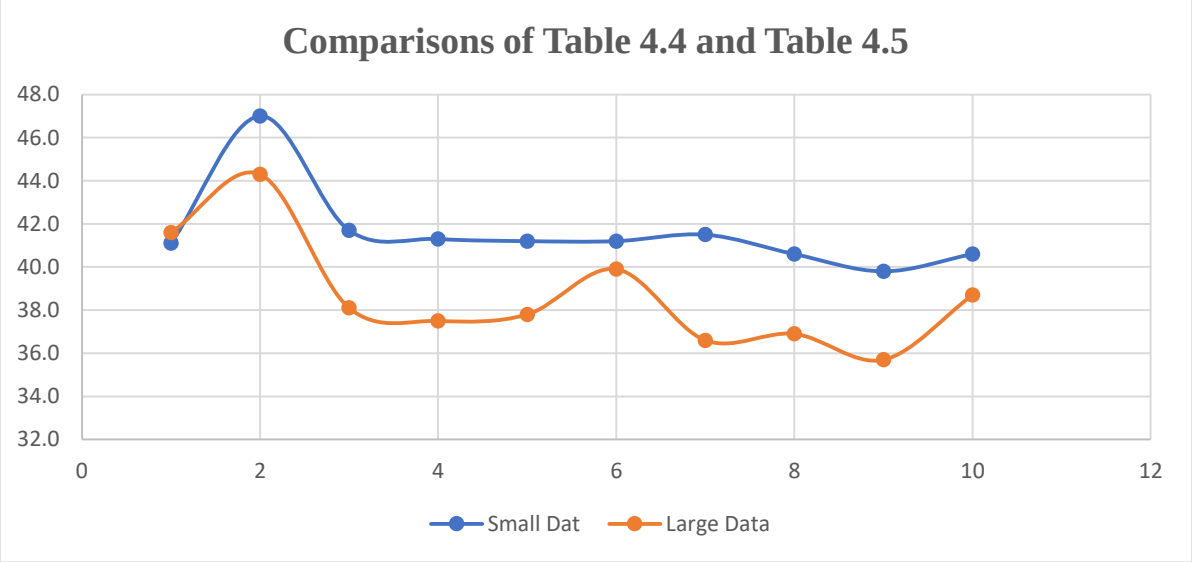


Figure 4.2: Comparisons of 48.89 and 83.13 hours data:

4.4 Testing with Larger Data:

In these below scenarios, Training data corpus is fixed it is not being changed but testing data is changing, to see variation in WER. I only used two different value of densities for these simulation training and testing 16 and 32. I skipped the value 8. Because now training data is crossing 83 hours of speech and for this much data 8 is not suitable as we can observe high WER against densities value 8 in Table 4.4 and Table 4.5

4.4.1 Scenario One

Table 4.6: 181 Speakers Training and 7 Speakers Testing

Sr#	Transcription Data (Hours)	No of Speakers	Testing Data (Minutes)	No of Speakers	Senones	Densities	Word Error Rate
1	83.13 h	181	178.095	7	4000	16	43.8% (12395/28314)
2	83.13 h	181	178.095	7	3000	16	43.5% (12325/28314)
3	83.13 h	181	178.095	7	2500	16	43.6% (12359/28314)
4	83.13 h	181	178.095	7	2000	16	44.2% (12523/28314)
5	83.13 h	181	178.095	7	1000	16	46.1% (13055/28314)
6	83.13 h	181	178.095	7	4000	32	43.9% (12424/28314)
7	**83.13 h**	**181**	178.095	**7**	**3000**	**32**	**43.10% (12192/28314)**
8	83.13 h	181	178.095	7	2500	32	43.11% (12208/28314)
9	83.13 h	181	178.095	7	2000	32	43.5% (12305/28314)
10	83.13 h	181	178.095	7	1500	32	43.3% (12268/28314)
11	83.13 h	181	178.095	7	1000	32	45.3% (12829/28314)

4.4.2 Description:

Eleven different simulations ran to complete this table. This Table 4.6 is a combination of different senones decreasing values from 4000 to 1000 and densities values 16 and 32. Same data set for training and testing yet we can observe WER difference about 2.2% also we can observe the best result we have is 43.1% while senones and densities values are respectively 3000 and 32 for that specific simulation. Total testing words are 28 thousand 3 hundred and 14.

Table 4.7: 181 Speakers Training and 7 Different Speakers Testing

Sr#	Transcription Data (Hours)	No of Speakers	Testing Data (Minutes)	No of Speakers	Senones	Densities	Word Error Rate
1	83.13 h	181	174.12	7	4000	16	46.4% (13052/28130)
2	83.13 h	181	174.12	7	3000	16	46.0% (12951/28130)
3	83.13 h	181	174.12	7	2500	16	46.4% (13041/28130)
4	83.13 h	181	174.12	7	2000	16	46.8% (13150/28130)
5	83.13 h	181	174.12	7	1000	16	49.0% (13792/28130)
6	83.13 h	181	174.12	7	4000	32	46.3% (13029/28130)
7	83.13 h	181	174.12	7	3000	32	45.6% (12841/28130)
8	83.13 h	181	174.12	7	2500	32	45.6% (12835/28130)
9	**83.13 h**	**181**	**174.12**	**7**	**2000**	**32**	**45.1% (12697/28130)**
10	83.13 h	181	174.12	7	1500	32	45.3% (12734/28130)
11	83.13 h	181	174.12	7	1000	32	47.7% (13412/28130)

4.4.3 Description:

This table contains same data as Table 4.6 but testing data is different here. Purpose of this to test if we change the testing corpus how much WER changes. Here we can see we have error rate with minimum value 45.1% to 47.7%. Again, whole tables WER varies between 2.6% difference. Here least values also have 2000 and 32 values of senones and densities respectively. Total testing words are 28 thousands 1 hundred and 30.

Table 4.8: 181 Speakers Training and 6 Speakers Testing

Sr#	Transcription Data (Hours)	No of Speakers	Testing Data (Minutes)	No of Speakers	Senones	Densities	Word Error Rate
1	83.13 h	181	169.56	6	4000	16	44.0% (11800/26814)
2	83.13 h	181	169.56	6	3000	16	44.2% (11857/26814)
3	83.13 h	181	169.56	6	2500	16	44.4% (11910/26814)
4	83.13 h	181	169.56	6	2000	16	44.4% (11892/26814)
5	83.13 h	181	169.56	6	1000	16	46.5% (12463/26814)
6	83.13 h	181	169.56	6	4000	32	43.5% (11677/26814)
7	83.13 h	181	169.56	6	3000	32	43.8% (11733/26814)
8	83.13 h	181	169.56	6	2500	32	43.4% (11629/26814)
9	**83.13 h**	**181**	**169.56**	**6**	**2000**	**32**	**43.4% (11623/26814)**
10	83.13 h	181	169.56	6	1500	32	43.8% (11755/26814)
11	83.13 h	181	169.56	6	1000	32	45.5% (12211/26814)

4.4.4 Description:

Here in this Table 4.8, everything is same as Table 4.6 and Table 4.7 but only testing data is changed here we have least WER 43.4% and most 45.5%. Which is between 2.1% of the difference. Least value of WER is again against the senones 2000 and densities 32. Which is same as Table 4.7. Here total testing words are 26 thousand 8 hundred and 14.

4.4.5 The average result of above three tables:

Table 4.9: Average results of 20 Speakers

Sr#	Transcription Data (Hours)	No of Speakers	Testing Data (Minutes)	No of Speakers	Senones	Densities	Word Error Rate (%age)
1	83.13 h	181	521.7838	20	4000	16	44.73333
2	83.13 h	181	521.7838	20	3000	16	44.56667
3	83.13 h	181	521.7838	20	**2500**	16	44.8
4	83.13 h	181	521.7838	20	**2000**	**16**	45.13333
5	83.13 h	181	521.7838	20	1000	16	47.2
6	83.13 h	181	521.7838	20	4000	32	44.56667
7	83.13 h	181	521.7838	20	3000	32	44.16667
8	83.13 h	181	521.7838	20	2500	32	44.03333
9	**83.13 h**	**181**	521.7838	**20**	**2000**	**32**	**44**
10	83.13 h	181	521.7838	20	1500	32	44.13333
11	83.13 h	181	521.7838	20	1000	32	46.16667

4.4.6 Description:

If we combine Table 4.6, Table 4.7 and Table 4.8 results it is 8 hours and 41 minutes of testing data. While average WER of all three tables can be seen in the last column of Table 4.9. Here as most of the previous tables, till we have 83-hour testing data again least values are against 2000 and 32.

4.4.7 Conclusion:

From these table results and comparison finally, I am on the point where I can say that for Urdu Speech Recognition system where you have 83 hours training data and a scenario like my data 2000 and 32 is the best combination for data training and testing.

4.5 Training and Testing while Language model includes testing corpus.

In this section, I am adding testing data in language model to see how it affects error rate. In below scenarios, I will be testing it with small and larger testing data to see how error rate got changed.

For now, least WER I have achieved is 35.7% in Table 4.5. So, I am testing it with from the same table.

Table 4.10: Testing data included in Language Model.

Sr#	Transcription Data (Hours)	No of Speakers	Testing Data (Minutes)	No of Speakers	Senones	Densities	Word Error Rate
1	83.13 h	181	44.89	5	4000	16	36.4% (2551/7018)
2	83.13 h	181	44.89	5	3000	16	35.5% (2494/7018)
3	83.13 h	181	44.89	5	2000	16	35.4% (2482/7018)
4	83.13 h	181	44.89	5	1000	16	38.3% (2685/7018)
5	**83.13 h**	**181**	44.89	**5**	**4000**	**32**	**35.6% (2496/7018)**
6	**83.13 h**	**181**	44.89	**5**	**3000**	**32**	**35.7% (2504/7018)**
7	**83.13 h**	**181**	44.89	**5**	**2000**	**32**	**33.7% (2363/7018)**
8	83.13 h	181	44.89	5	1000	32	37.1% (2600/7018)

4.5.1 Description:

In this Table 4.10, I have included same data, as Table 4.5 but the difference is that the language model is different here. This language model also contains testing words in it. So here we can observe we have different results from Table 4.5. Even there is the lowest value of WER which is 33.7%. In Table 4.5 which is 35.7%. A complete comparison of both tables is shown in Chart 4.3. If we compare both tables as it is done in Chart 4.3. there is the difference in WER from 1% to 2.7% before adding testing data to after adding testing data. I have tested it with larger data too. This is explained below.

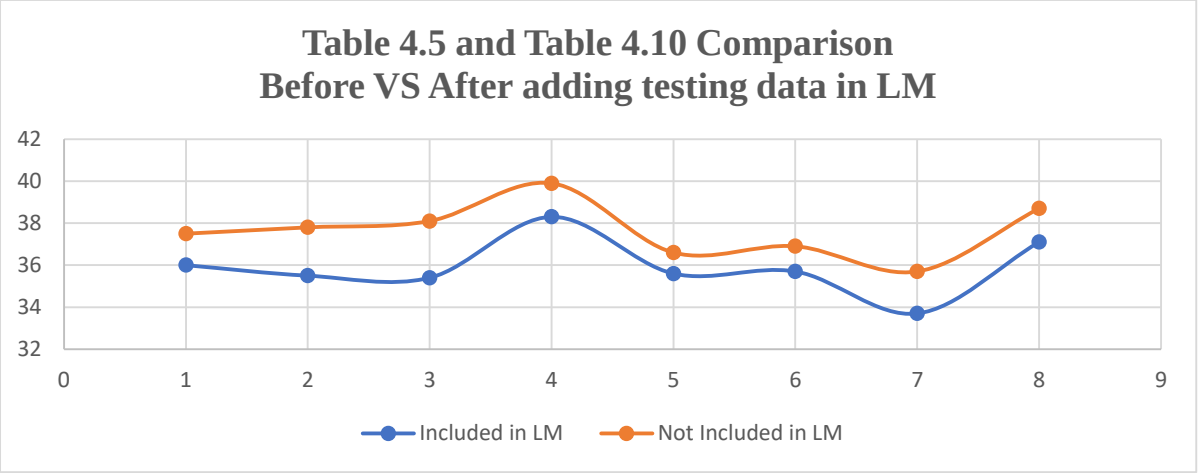


Figure 4.3: Table 4.5 and Table 4.10 Comparison

4.5.2 With Larger testing data:

Table 4.11 Testing Data Included in LM With Larger Data

Sr#	Transcription Data (Hours)	No of Speakers	Testing Data (Minutes)	No of Speakers	Senones	Densities	Word Error Rate
1	83.13 h	181	169.56	6	4000	16	42.3% (11353/26814)
2	83.13 h	181	169.56	6	3000	16	42.6% (11436/26814)
3	83.13 h	181	169.56	6	2500	16	42.6% (11422/26814)
4	83.13 h	181	169.56	6	2000	16	42.7% (11449/26814)

5	83.13 h	181	169.56	6	1000	16	44.5% (11924/26814)
6	83.13 h	181	169.56	6	4000	32	42.2% (11307/26814)
7	83.13 h	181	169.56	6	3000	32	41.8% (11218/26814)
8	**83.13 h**	**181**	**169.56**	**6**	**2500**	**32**	**41.3% (11071/26814)**
9	83.13 h	181	169.56	6	2000	32	41.7% (11186/26814)
10	83.13 h	181	169.56	6	1500	32	42.0% (11269/26814)
11	83.13 h	181	169.56	6	1000	32	43.8% (11741/26814)

4.5.3 Description:

Table 4.11 have all same values as Table 4.10 except testing data is larger than Table 4.5 and Table 4.10. Here we have testing data used in Table 4.8. Here we are observing lowest value 41.3% which was 43.4% in Table 4.8. A complete comparison of both tables is shown in Chart 4.4. We can observe difference is between both table's WER is 1.3% to 2.1%.

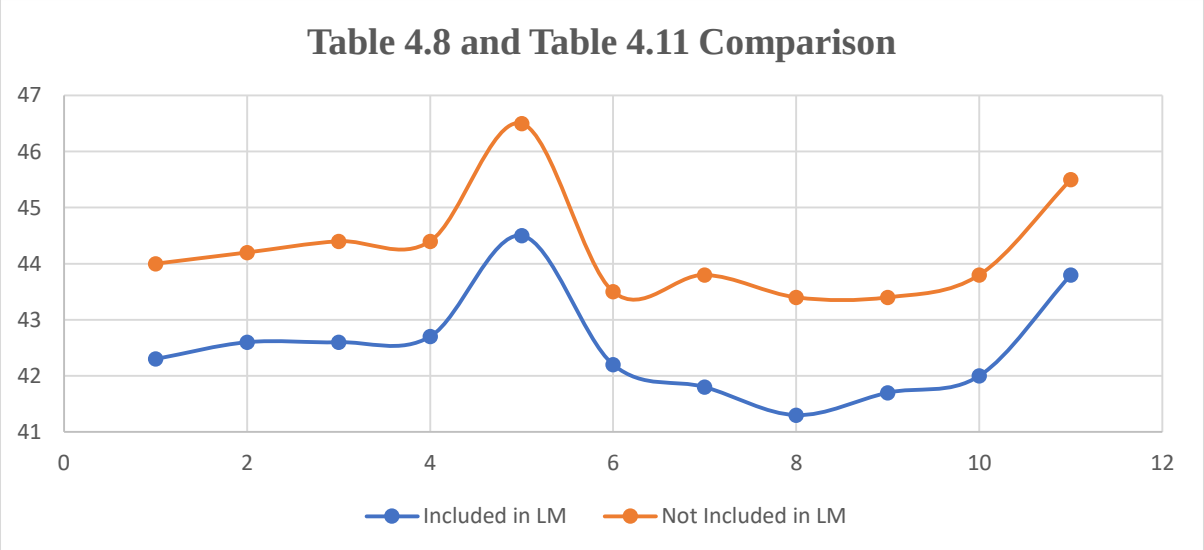


Figure 4.4: Table 4.8 and Table 4.11 Comparison

4.6 Conclusion:

In this chapter training and decoding have been discussed in details. From all these experiments it is found if we increase training data accuracy increases, as my data increased from 23 hours to 48 and then 83 hours. Another founding is accuracy is not only about getting the least number but it is also affected by changing data. If we change data even exactly a same number of words are in both data. Still there WER will be different. Because some person may not have spoken well or another reason can be if test data is not clean. About testing data status I am discussing it in details.

Chapter 5

5 Discussion and Findings

5.1 Application to Compare Transcription and results:

While running data training and decoding and getting different results, I was curious about reducing the WER because it makes the system more accurate and better. But hurdle was to identify what is error and what is not and which are the words which become errors. So I wrote some codes and then merged them and made an application to compare errors. From literature review of Urdu speech recognition, no one has done this type of work. So I think this can give a new dimension to reduce errors and increase accuracy.

Image 5.1 is a front view of the application. It contains eight buttons which has self-explanatory names still these are being defined below. Then there is text area which acts as a log. All activities are shown in written form here. Below this, there is a table which is to compare words. This application does multiple jobs. Which I am describing one by one.

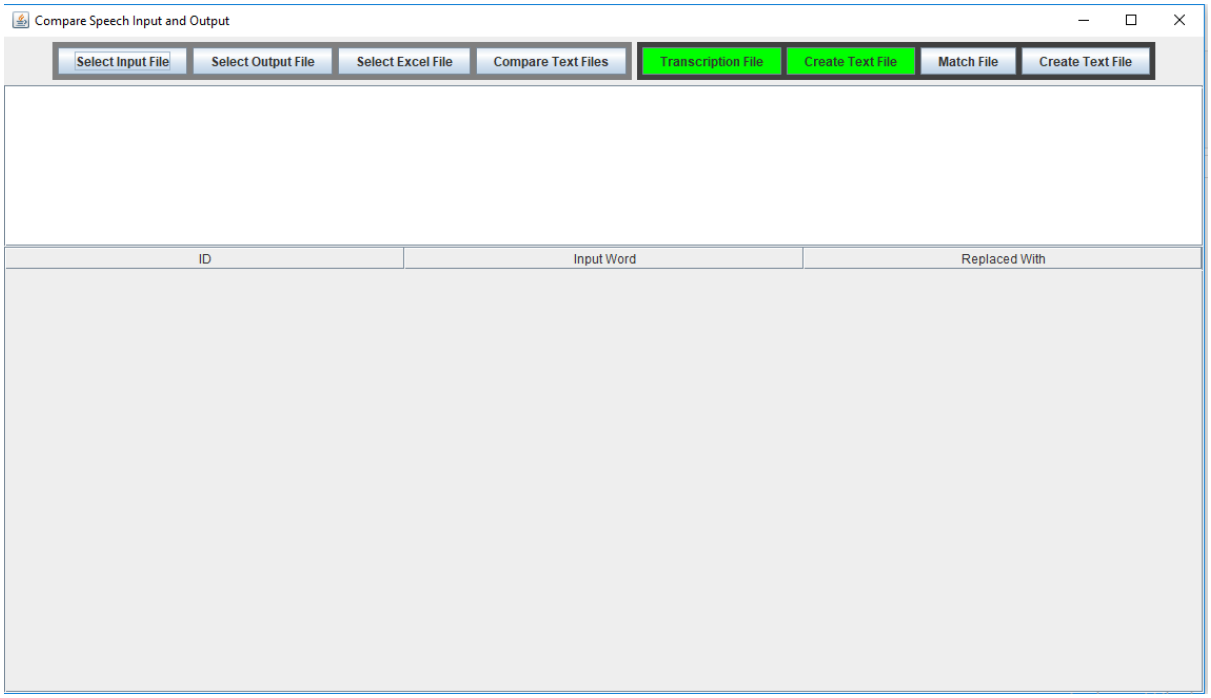


Figure 5.1: Home Screen

5.1.1 Transcription file to text file:

Transcription file is in a specific format. <s> text</s> (path). But to compare it with output file we only need text. So this step is to convert transcription file to text file. For this purpose, there are two green buttons. Click Transcription file button and a dialog box will be opened and will expect a transcription file as shown in Figure 5.2. Select file and press create a text file as shown in Figure 5.3.

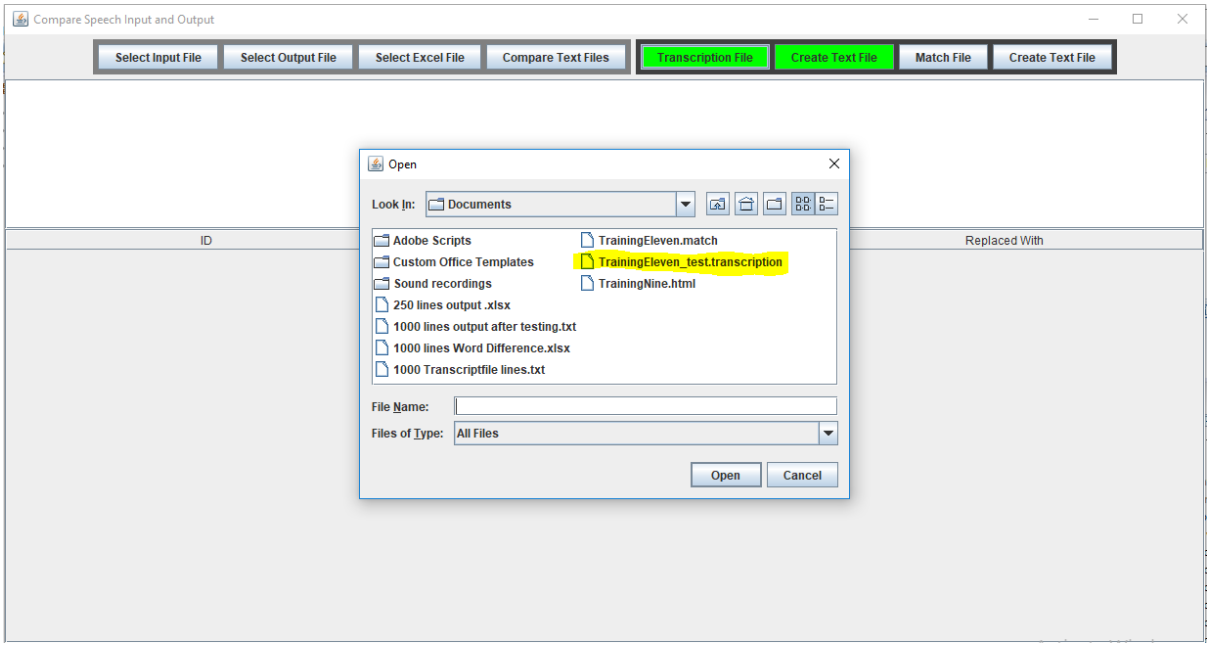


Figure 5.2: Transcription file Selection

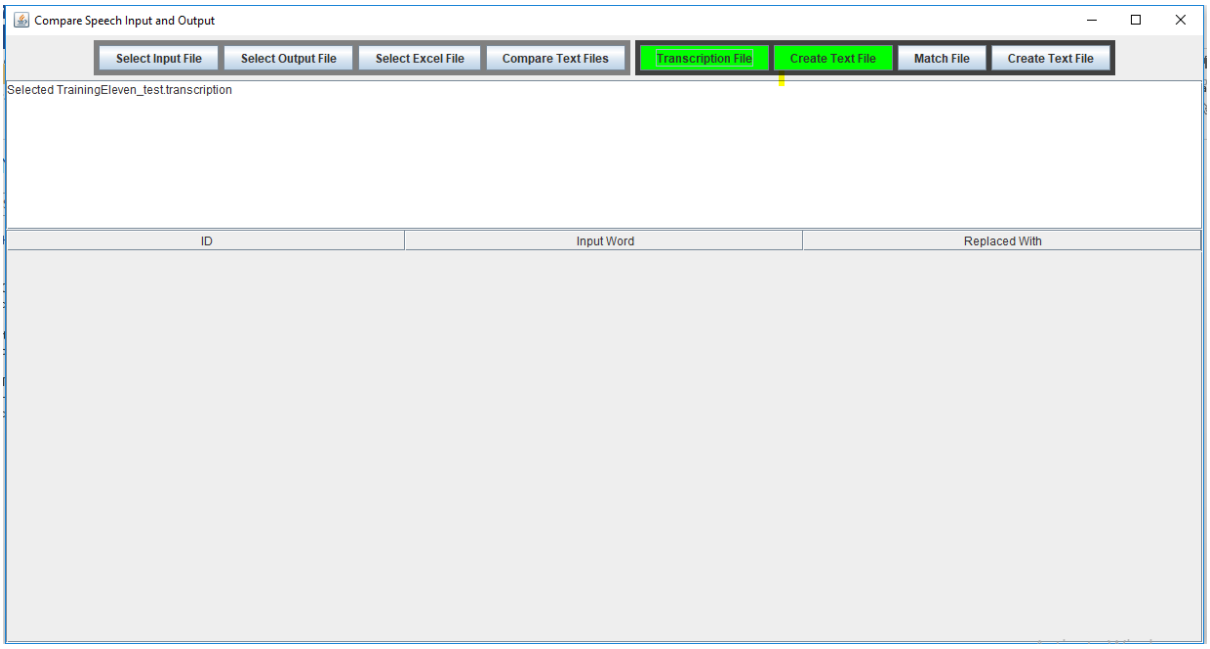


Figure 5.3: Transcription File Selected

As a result, a text file will be created in the same folder from where transcription file is chosen. Now, this is a text file which will be used as input to compare another file. Which is described in Figure 5.4.

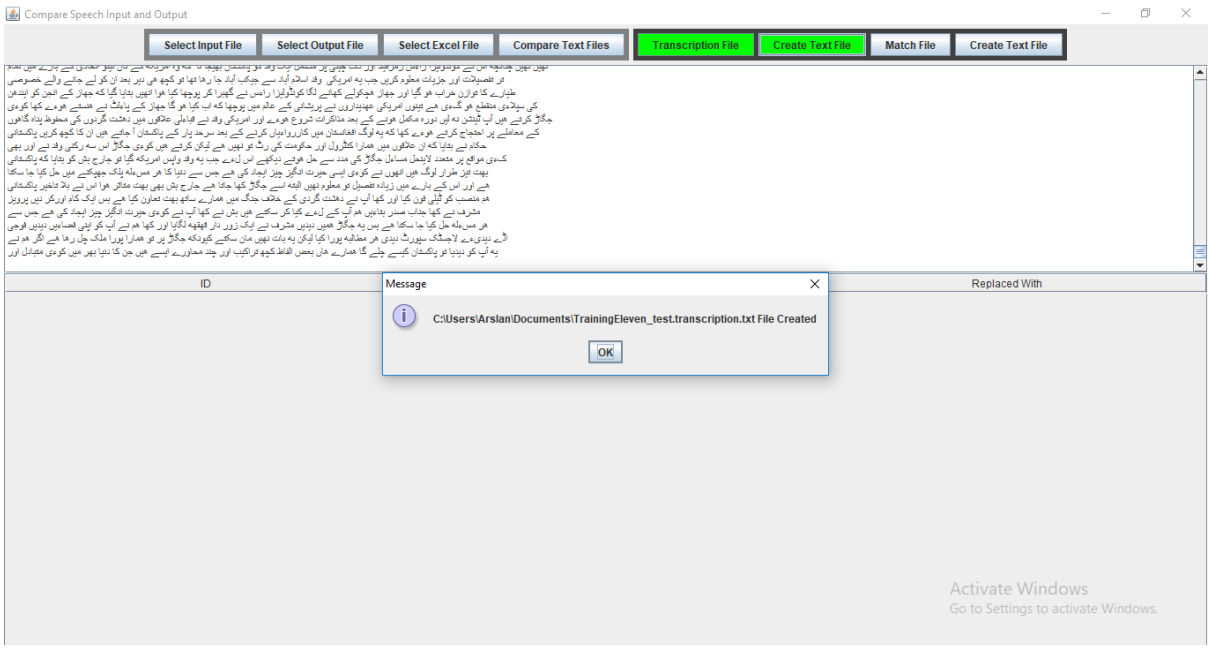


Figure 5.4: Simple Text File Created From Transcription File

5.1.2 Match file to Text File:

Match file is output file which is produced after decoding process. This file contains the identified text which is converted from speech to text in decoding step of CMUSphinx. Its format is “text (speaker id)”. Again from this file we only need text so. Click on Match file button and select file as step one and as shown in Figure 5.5. The selected file name will be shown in log portion as shown in Figure 5.6.

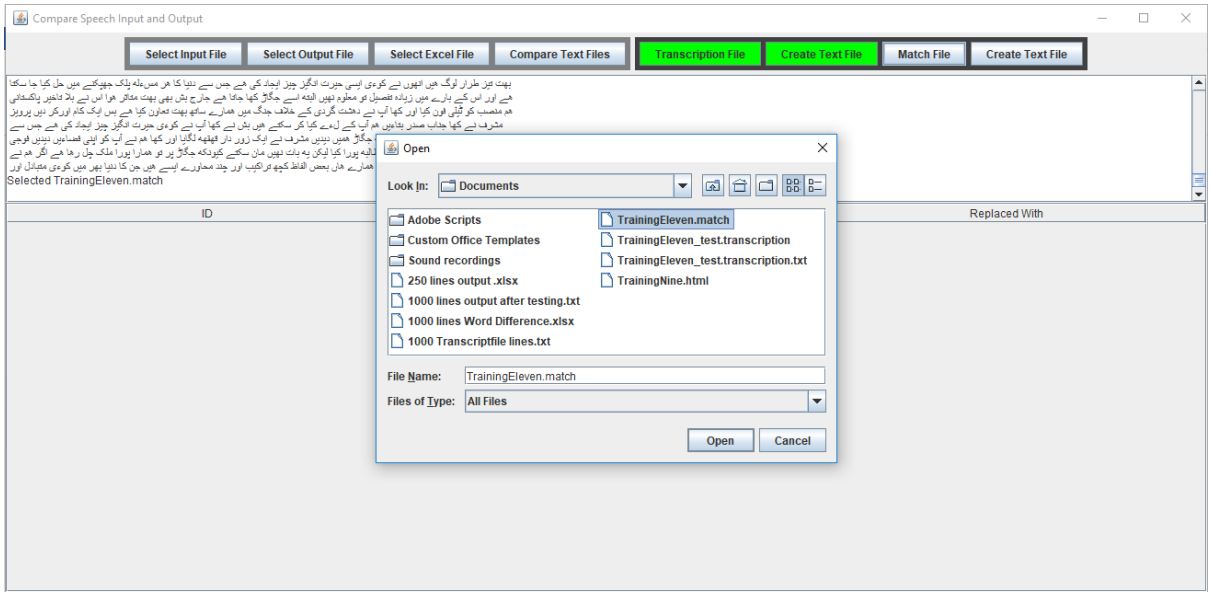


Figure 5.5: dot Match File Selection

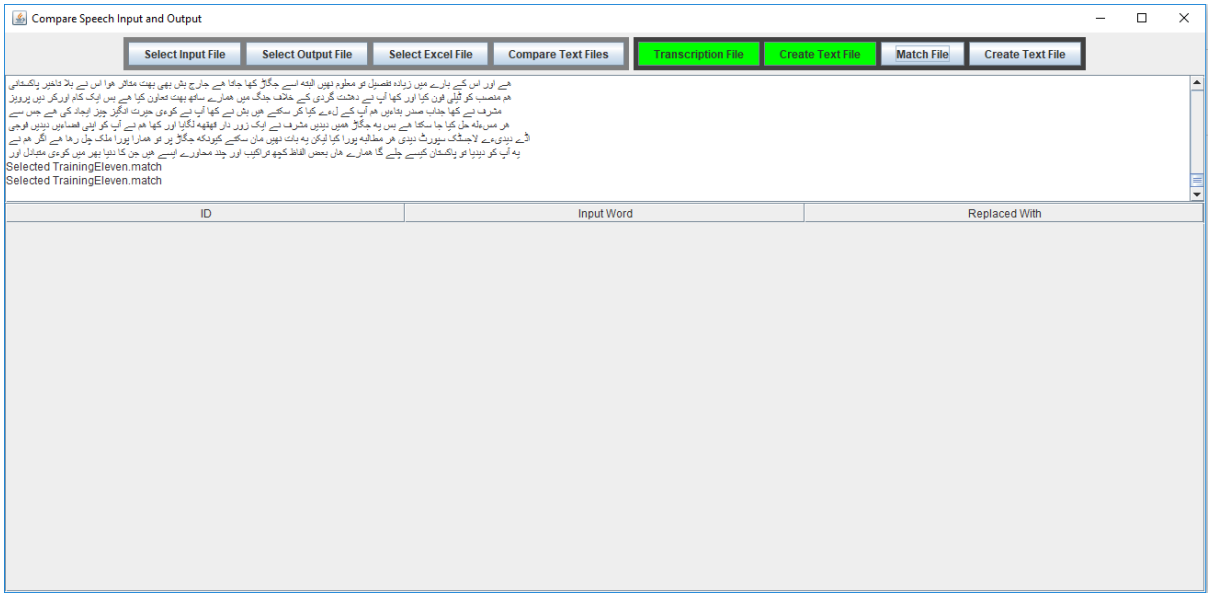


Figure 5.6: dot Match File Selected

Now click on Create text file button located on left side of Match file button and in few seconds (depending on file size and computer specifications) file with the same name but in txt extension will be created as shown in Figure 5.7. This is output file which will require as a second entity to find differences of input and output.

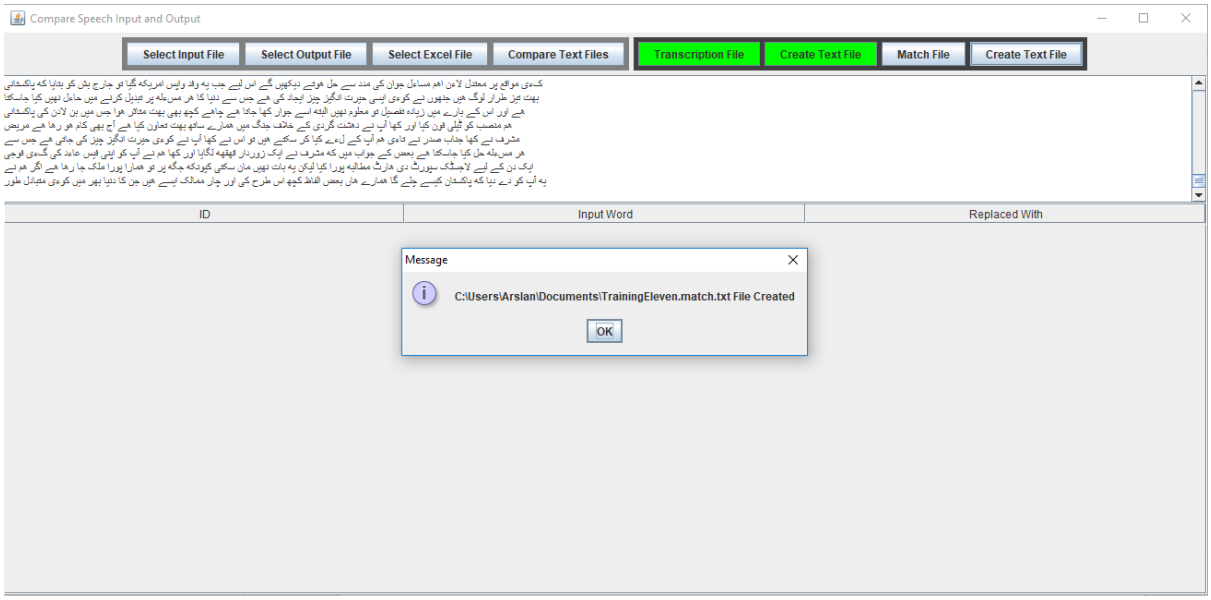


Figure 5.7: dot Match File to Text File Creation

5.1.3 Compare Input and Output Files:

Now we have two files form each previous step. The first file is input file which is transcription test file and second is the output file, produced after decoding. Another file we need is excel file so we can save comparison of words. A table can be seen in All Images it is to show the output of compared words but these values cannot be saved from the table so as a replacement I created an excel file option for further analysis. First, three buttons take the input of these above-described files and the fourth button is to compare these text files. Figure 5.8 shows that all three files are selected.

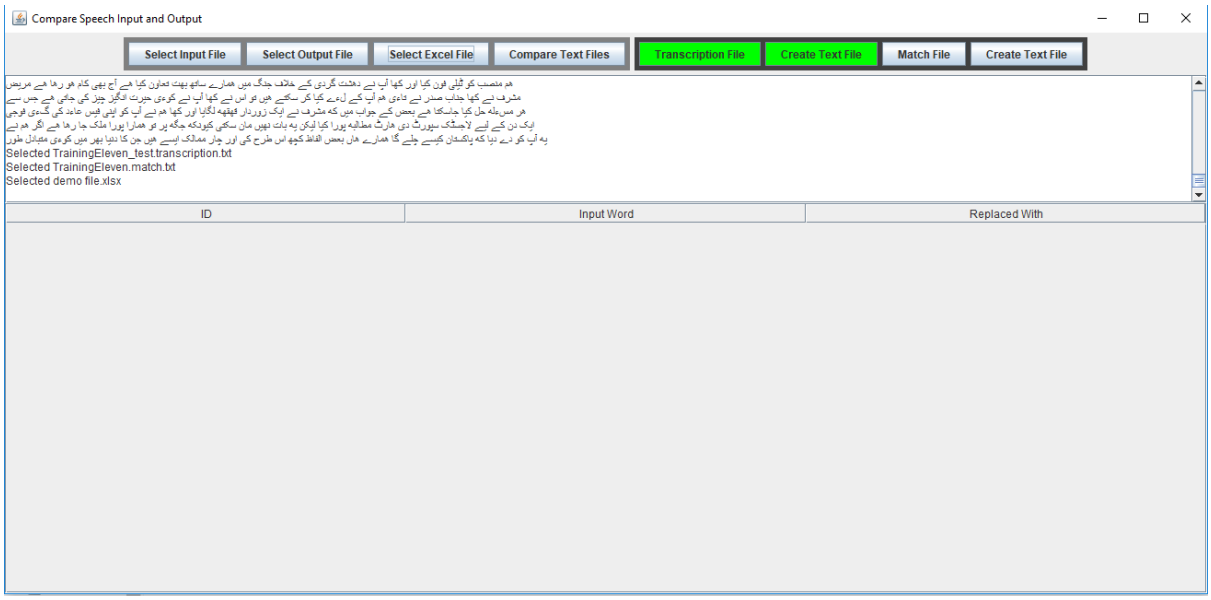


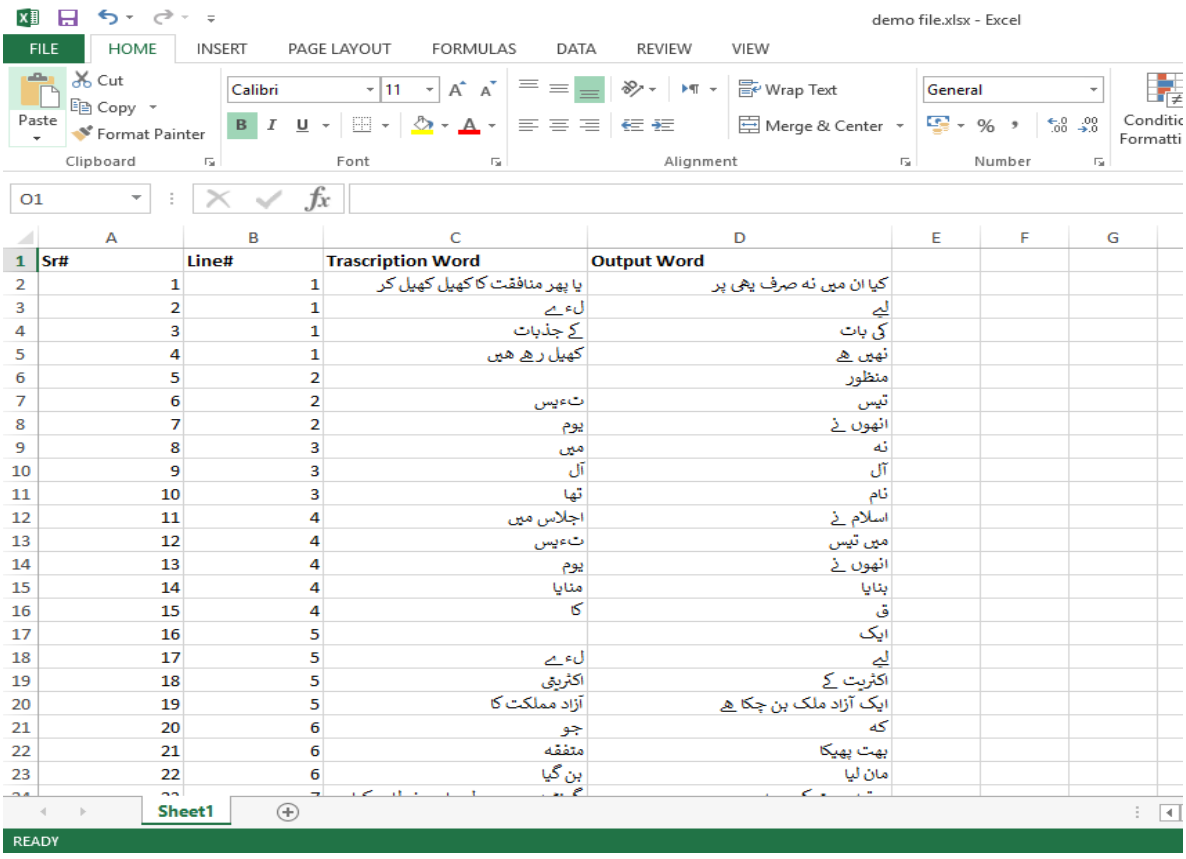
Figure 5.8: Excel Output File Selection

After Clicking the compare text files button system will take some time depending on file sizes and computer specifications and as output excel file will be filled with data as shown in

Figure 5.10 it has 4 columns, First is serial number, second is line number which is taken from files, third is transcribed words and forth is decoded words. There is a table in software that will also be filled as shown in Figure 5.9 as ID it shows the line numbers of text files.

ID	Input Word	Replaced With
1	یا پھر منافقت کا کھیل کھیل کر	کیا ان میں نہ صرف یہی پر
1	ل ءے	لیے
1	کے جذبات	کی بات
1	کھیل رھے ھیں	نہیں ھے
2		منظور
2	ت ءیس	تیس
2	یوم	انھوں نے
3	میں	نہ
3	ال	ال
3	تھا	نام
4	اجلاس میں	اسلام نے
4	ت ءیس	میں تیس
4	یوم	انھوں نے
4	منایا	بنایا
4	کا	ق
5		ایک
5	ل ءے	لیے
5	اکثریتی	اکثریت کے
5	ازاد مملکت کا	ایک ازاد ملک بن چکا ھے
6	جو	کہ
6	متفقہ	بہت پھیکا
6	بن گیا	مان لیا
7	گھنٹوں پر محیط جامع خطاب کیا جو	مرتبہ موت کی وجہ
7		پر کیا جو ان
7	تجربے اور	کلچر میں ہو

Figure 5.9: Final Output Results



Sr#	Line#	Trascription Word	Output Word
1	1	یا پھر منافقت کا کھیل کھیل کر	کیا ان میں نہ صرف یہی پر
2	1	ل ءے	لیے
3	1	کے جذبات	کی بات
4	1	کھیل رھے ھیں	نہیں ھے
5	2		منظور
6	2	ت ءیس	تیس
7	2	یوم	انھوں نے
8	3	میں	نہ
9	3	آل	آل
10	3	تھا	نام
11	4	اجلاس میں	اسلام نے
12	4	ت ءیس	میں تیس
13	4	یوم	انھوں نے
14	4	منایا	بنایا
15	4	کا	ق
16	5		ایک
17	5	ل ءے	لیے
18	5	اکثریتی	اکثریت کے
19	5	آزاد مملکت کا	ایک آزاد ملک بن چکا ھے
20	6	جو	کہ
21	6	متفقہ	بہت پھیکا
22	6	بن گیا	مان لیا

Figure 5.10: Output Excel File

So here is my small but helping application which I used, to find the error and tried to reduce them in order to improve accuracy. Now, these errors are being described in details in next sections.

5.2 Findings from experiments.

There are some different types of errors which we can observe in here from software output. These are too much error and cannot be added here as it will increase pages of the thesis. So I have included few here in order to describe the problem.

Few mix errors are shown in Table 5.1

Table 5.1: Overall Errors

Transcription word	**After Testing replaced with**	**Transcription word**	**After testing Replaced with**
لۓ	لیے	**خود بخود**	خودبخود
آل	آل	**آدم**	آدم
کا	ق	**سدا**	صدا
اس لۓ	کس لیے	**بھلاتا**	بلاتا
مطالعے	مطالعه	**آراءیوں**	آراءیوں
مطالعے	مطالعه سے	**آبادی**	آبادی
کۓ	کیے	**فی صد**	فیصد
قرآن	قرآن	**جا رھی**	جارھی
کر سکتے	کرسکتے	**نوازشریف**	نواز شریف
انھوں	انھوں	**آتے**	آتے

نشونما	**نشوونما**	نا گزیر	**ناگزیر**
شہباز شریف	**شہبازشریف**	کہ	**کے**
آسان	**آسان**	کے	**کی**
طلبا	**طلبہ**	کرسکتا	**کر سکتا**
واحترام	**و احترام**	کا	**کہ**
سرگرمیوں	**سر گرمیوں**	اسے	**اس سے**
آف	**آف**	پانامہ	**پاناما**
آ	آ	آیں	**آیں**
عملدرآمد	**عمل درآمد**	آرہا	**آ رہا**
آیا	**آیا**	وزیر اعظم	**وزیراعظم**
ء		جاسکتی	**جا سکتی**
دوا ساز	**دواساز**	انہیں	**انہیں**
آمادہ	**آمادہ**	آیی	**آیی**
خود کش	**خودکش**	سیاستدانوں	**سیاست دانوں**
آپ	**آپ**	آور	**آور**
آیی	**آیی**	کردیا	**کر دیا**
سکیورٹی	**سیکورٹی**	دھشتگردوں	**دھشت گردوں**
جاسکتا	**جا سکتا**	آیندہ	**آیندہ**

آپریشن	آپریشن	**دیدیا**	دے دیا

We can categorize these errors as

1- Actual Errors.
2- Errors due to space
3- Errors due to replaced character.
4- Error due to alternative error.
5- Incorrect words

5.2.1 Actual Errors:

These are types of error which are errors and only reason can be of these errors is either speaker did not speak these words correctly, or there is noise in surroundings. It can be observed mostly this type of error or either start or end of the recordings. This type of errors are too many, some of them are given in Table 5.2

Table 5.2: Actual Errors

Transcription word	**After Testing replaced with**	**Transcription word**	**After testing Replaced with**
مساءل حل کرنے کی	ان حالات میں یہ	**تءیس**	تیس
پوچھا	کھینچا	**یوم**	انھوں نے
مطالبہ	تازہ	**تجربے اور**	کلچر ڈے ھو
کا مطلب	ان میں	**عدل**	عدم
سبق قوم کے ذھن	سبب بنتے رھے ھیں	**ایمان نظم**	ان منظم
مقام یا	مطالعہ کیا	**ایمان**	انسان
کاروبار ریاست	کارروایٔ سیاست	**پاک**	بات
بچ جاتیں	بات کی جاتی ھے	**ملزمان**	وہ زمانہ

کے	که	**کا ٹراٶل**	هار پرتگال میں
هی	ایسی	**لڑکیوں**	لڑکی هوں
اضافه	اضافے	**دودھ**	دور دور

5.2.2 Errors due to space:

These are types of errors which are due to a space character. Like in testing data there is a combination of two words **خود بخود.** But it is identified as خودبخود. Which is a single word. But in reading there is no problem. If someone is giving dictation, then it will not be considered as an error. Other examples of these types of error are given in Table 5.3

Table 5.3: Errors Due to Space

Transcription word	**After Testing replaced with**	**Transcription word**	**After testing Replaced with**
کر سکتے	کرسکتے	**خود بخود**	خودبخود
جا رهی	جارهی	**کر سکتا**	کرسکتا
نوازشریف	نواز شریف	**وزیراعظم**	وزیر اعظم
ناگزیر	نا گزیر	**جا سکتی**	جاسکتی
عمل درآمد	عملدرآمد	**دواساز**	دوا ساز
سیاست دانوں	سیاستدانوں	**خودکش**	خود کش
شهبازشریف	شهباز شریف	**جا سکتا**	جاسکتا
کر دیا	کردیا	**دیدیا**	دے دیا
دهشت گردوں	دهشتگردوں	**سر گرمیوں**	سرگرمیوں

و احترام	واحترام	**فی صد**	فیصد
روز مرہ	روزمرہ	**بھر حال**	بھرحال
سرگرم	سر گرم	**کیلۓ**	کے لیے
پیشرفت	پیش رفت	**طورپر**	طور پر
ہو چکا	ہوچکا	**نا کردہ**	ناکردہ
پرامن	پر امن		

5.2.3 Errors due to replaced character:

There are some words in the Urdu language which are correct in both ways while we are reading. Like بے کار and بیکار are correct in both ways. It is because of the behavior of ے which is a nonjoiner and not being joined with next word and in بیکار it is replaced with Choti yayي [401]. But for the system, it is either a combination of two words or one word. So it becomes an error if the come against each other. Same is the case with **دیدیا** and دے دیا. There are also some words whose Unicode value is different in some keyboards like ک ، ي ، ی ،ہ etc.[401]

These types of errors are due to alternative but correct character. Like in Urdu **لۓ and** لیے are both correct words. But for computer/system, these both are different words. As these two are a combination of **ل, ۓ** and **ے** But لیے is a combination of ی ، ل and ے **.** Another example of this is **انھوں** combination of **ں و ہ ن ا** and the انھوں combination of ں و ھ ن ا.

Another common reason for this error is آ. It is in multiple words. These all words become the reason for increasing WER when these are not identified as correct words. آ have two types. One is single character آ but other is a combination of two characters ا and ٓ. Table 5.4 has a list of this type of words.

Table 5.4: Errors due to replaced character

Transcription word	**After Testing replaced with**	**Transcription word**	**After testing Replaced with**
آراءیوں	آراءیوں	**آف**	آف

آبادی	آبادی	**آیا**	آیا
آءیں	آءیں	**آ**	آ
آ رھا	آرھا	**عمل درآمد**	عملدرآمد
آءی	آءی	**آمادہ**	آمادہ
آور	آور	**آپ**	آپ
آتے	آتے	**آءی**	آءی
قرآن	قرآن	**آسان**	آسان
آءندہ	آءندہ	**آپریشن**	آپریشن
لءے	لیے	**انھوں**	انھوں
طلبہ	طلبا	**انھیں**	انھیں
دیءے	دیے	**چاہءے**	چاھیے
کءے	کیے	**آسمان**	آسمان

5.2.4 Error due to the alternative word:

There are some words which sound almost same. These words also caused errors rate to increase but these words are actual error and it is hard to replace these words. Quantity of these words is very less, some of them are shown in Table 5.5

Table 5.5: Error due to alternative word

Transcription word	**After Testing replaced with**	**Transcription word**	**After testing Replaced with**

بھلاتا	بلاتا	**سدا**	صدا
مخصوص	محسوس		

5.2.5 Incorrect words:

There are some errors which are due to incorrect words. There are some characters which sound the same but are actually different like "ز" and "ذ" are mostly confused with each other even in newspapers [401]. Words like پذیر، عزیز are being affected by this. So incorrect character leads to the incorrect word and higher WER. On another hand, there are some words which are incorrectly written in corpus either in training or testing. Some of the words are given in Table 5.6

Table 5.6: Incorrect words

Transcription word	**After Testing replaced with**	**Transcription word**	**After testing Replaced with**
نشوونما	نشونما	**سیکورٹی**	سکیورٹی
وڈیو	ویڈیو	**پروسیس**	پراسس
نقطه	نکته	**کیسلی**	کسیلی

5.3 Conclusion:

This chapter contains a comparison of different types of errors and details of errors. A tool is created to compare input or spoken text and recognized text or output text. With this tool, anyone who uses CMUSphinx for data training and decoding can test his input and output. Most of the work is converted to automated work. The purpose was to convert manual analysis work to automatic so error rate can be reduced and system accuracy can be improved. The tool is made in Java and easy to use. Step by step details has been given in this chapter.

Chapter 6

6 Speech Recognition Based Automated Technical Support System

6.1 Introduction:

Speech recognition is being used in multiple applications available at android store like “Speech Assistant AAC[53]”, “Speechnotes Speech To Text [54]”, “Speech To Text [55]” , “Speech to Text Translator TTS [56]”, “Text and Drive [57]” etc... These applications serve their purpose for what they are created. First, these all are based on the English language. I made two different apps to show the results and use of Urdu speech recognition. First is a call center base technical support system. Which guides the caller to solve his internet communication problem. It's dialogue based application where system ask a question and user replies according to this question. If the answer is as expected it moves to next step, if there is some problem in identification or answer is not as expected. The system requires a correct answer to continue. After the end of the process either internet will be working properly or complaint will be recorded. Another app is also using Urdu speech recognition, it have two main modules first is command and control and the second is where the user speaks and type message, then this message can be shared to other apps and also can be copied to the clipboard for further use. Both applications are being discussed in details here with the help of screens shots.

6.2 Call Center App (Automated Technical Support System):

This application is targeted for an audience which wants to submit a complaint about their internet dis-connectivity. It is demo app which can be installed on Android mobiles to test. Figure 6.1 shows the flow of the application. The system starts with saying “ کال سنٹر میں خوش آمدید۔ اپنی شکایت بیان کریں” and then user responds. All green background rectangles are when system speaks and blue background are expected answers from the user.

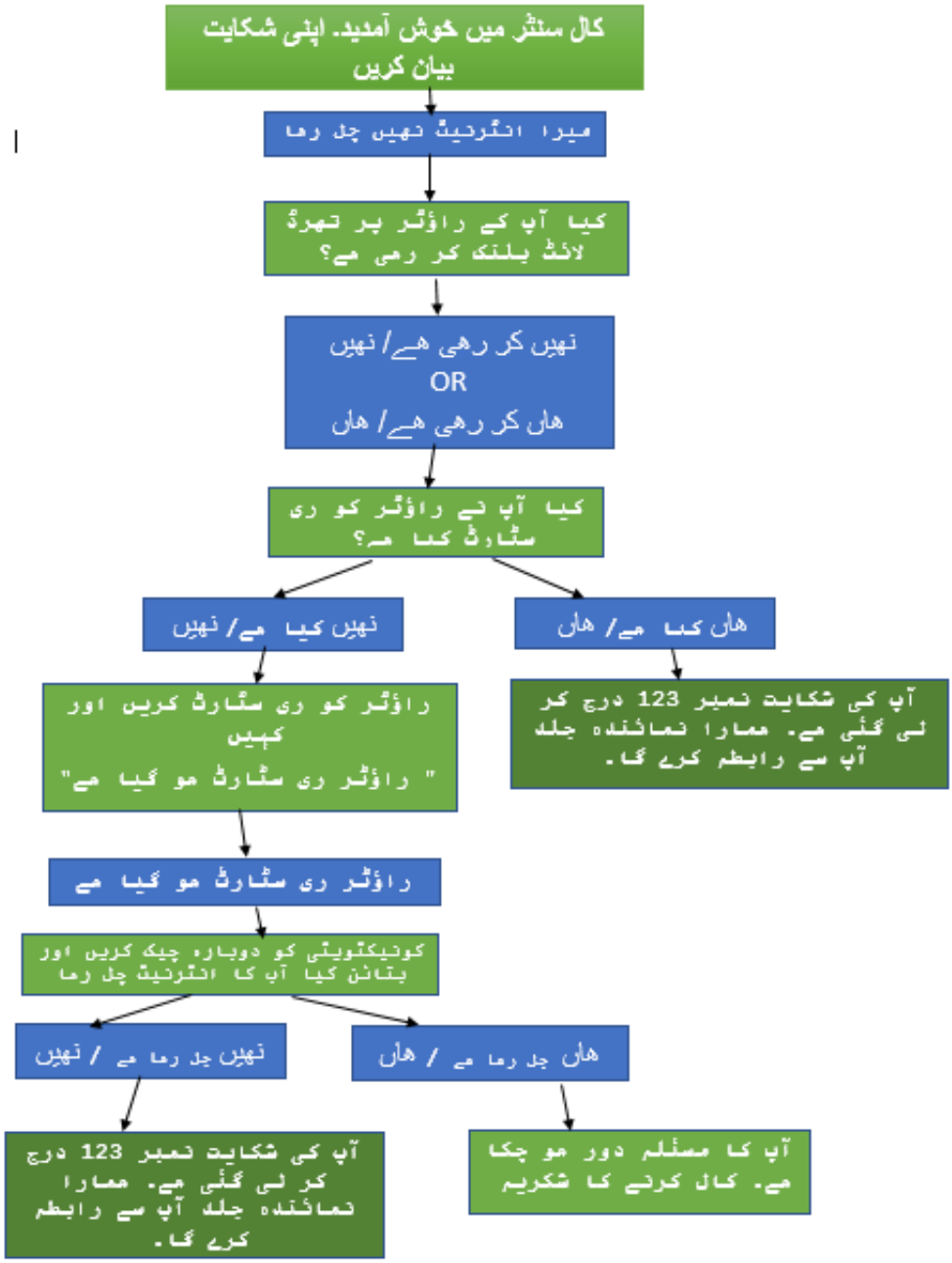


Figure 6.1: Flow of Application

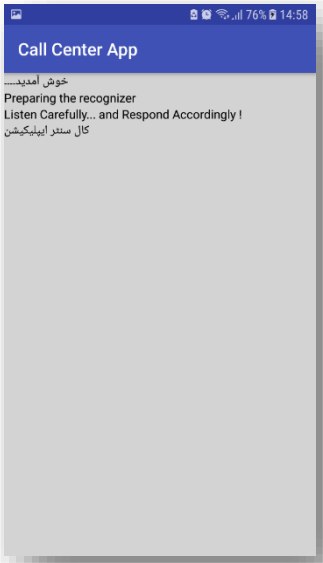


Figure 6.2: Home Screen

Figure 6.2 is home screen of the application. When app Icon is tapped then this view will be opened. Front screen is shown and as a background process in this time recognizer loads its required resources.

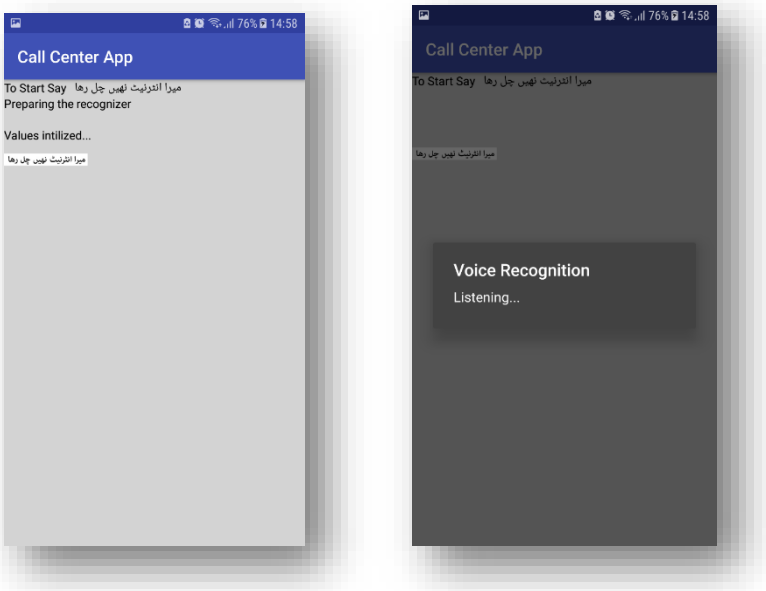


Figure 6.3: Resource loading,

Figure 6.4: Waiting for user to speak

Figure 6.3 shows the same screen but in this step, values are changed. In this step system speaks "کال سنٹر میں خوش آمدید۔ اپنی شکایت بیان کریں“After this user is supposed to narrate his problem, which in this case is

میرا انٹرنیٹ نہیں چل رہا | نیٹ میں پرابلم ہے | انٹرنیٹ خراب ہے | انٹرنیٹ میں پرابلم ہے | نیٹ میں پرابلم ہے “
”| نیٹ نہیں چل رہا | نیٹ خراب ہے | انٹرنیٹ کونیکٹ نہیں ہو رہا | انٹرنیٹ کونیکشن نہیں ہو رہا

But for the view I have shown only one line because it is demo app. The user must have to say any line from these sentences there are OR sign between these sentences if user speaks any of this line it will be identified. Figure 6.4 is next step of Figure 6.3. It started when system ends its speech and expect the user to speak. In this step, the user shall have to say any line to go to next step.

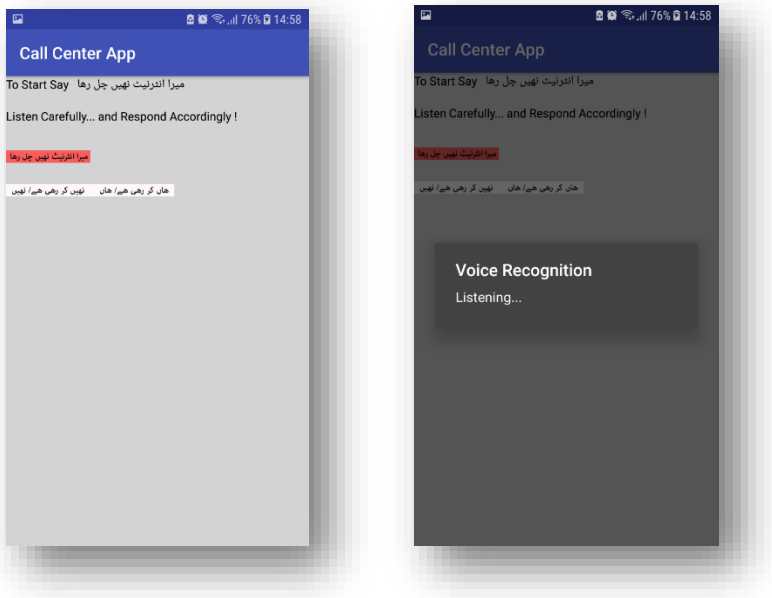


Figure 6.5: First line identified

Figure 6.6: Waiting for next input

Figure 6.5 is when already it is identified it changes to a red background. When the background is white it means the line is not identified. After identification of first step system speaks “ کیا آپ کے راؤٹر پر تھرڈ لائٹ بلنک کر رہی ہے؟” here the system is querying about that third light of router is blinking or not, in response there can be two answers that yes it is blinking or no it is not. To guide here two further lines are shown “نہیں کر رہی ہے/ نہیں” and “ہاں کر رہی ہے/ ہاں” where user have to reply between these two lines. In Figure 6.6 it is a screen where the user is speaking and the system is expecting some speech input. Here a user must have to answer between these two lines only one of these answer will be accepted. If user replies with any other option he will be shown as Figure 6.14

Figure 6.7: Query about Restarting Router

Figure 6.8: Reply about Restarting Router

After the previous step now in Figure 6.7 user have replied with "ہاں کر رہی ہے" or "ہاں" that's the reason its background turned red when this line is identified now its turn for the system to speak. Now system has to confirm that if user has restarted his/her router or not because restarting the router sometimes solve the problem as if there is a minor problem. So in this step system is speaking "کیا آپ نے راؤٹر کو ری سٹارٹ کیا ہے؟". In response to this, there are two possibilities that either he/she have done already or not. So that's what the answer, system is expecting which is shown in Figure 6.8. Now in this screen user have spoken yes he has restarted the router and in Urdu, it is replied with either "ہاں کیا ہے" or "ہاں" and its background turned red. So After this step, we are now in this condition that user's internet is not working and he has restarted his/her router. Now it's time to register his/her complaint. So after this step his/her complaint will be registered and he will be assigned a complaint ID for further queries. From these easy steps without call center's employ involvement user have registered his/ her complaint.

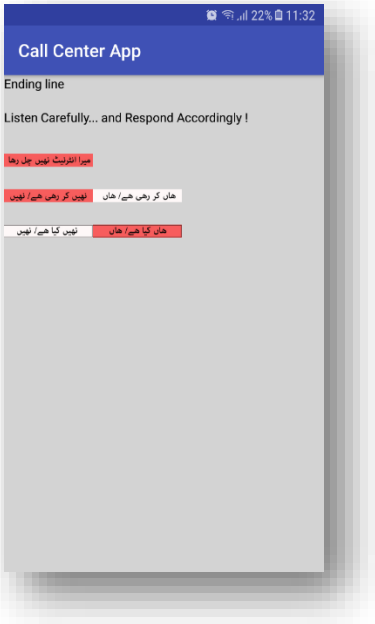


Figure 6.9: Query about restarting router

This Figure 6.9 is different from Figure 6.7 although in this step system will be asking the same question that if the user has restarted the router or not. In this step, his router's third light is not blinking. As this option's background is red. Now the user has to reply whether he has restarted his router or not. If the user replies that he has restarted his router then his complaint will be register and complaint ID will be assigned to him. As explained in the previous step.

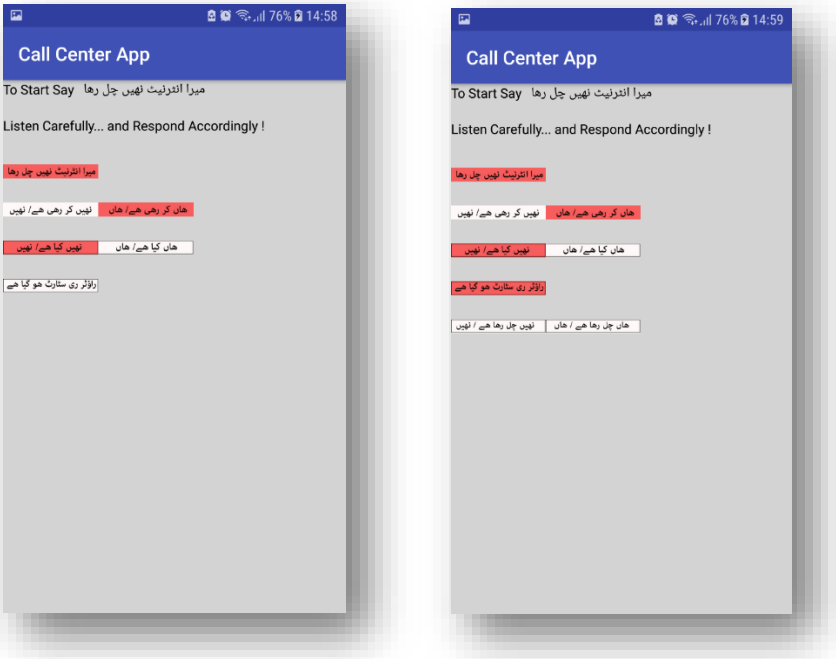


Figure 6.10: Waiting for Router Restart

Figure 6.11: Next Step after Router Restart

In Figure 6.10, it is further but the alternative step of Figure 6.7, Figure 6.8 and Figure 6.9 where user replied with "نہیں کیا ہے/ نہیں" that he had not restarted his/her router. Now the system will guide him/her further and ask him to restart the router. This is the line that system will speak "راؤٹر ری سٹارٹ ہو گیا ہے " راؤٹر کو ری سٹارٹ کریں اور کہیں". There is only this line in this step, User must have to reply that he has restarted the router. While it is user dependent he actually restarts the router and says he has done it or only say.

In Figure 6.11 this screen shows that "راؤٹر ری سٹارٹ ہو گیا ہے" background have turned red. It means the user has spoken this line and it has been identified correctly. Now the system will make a query from a user by saying "کونیکٹویٹی کو دوبارہ چیک کریں اور بتائں کیا آپ کا انٹرنیٹ چل رہا ہے". Check the connectivity again and reply that your internet is working or not. After system finish his speech now user have to respond from shown two options "ہاں چل رہا ہے / ہاں" in this case user restarted router and his internet communication got recovered and another option is " نہیں چل رہا ہے / نہیں" that he restarted the router but his internet is still not working.

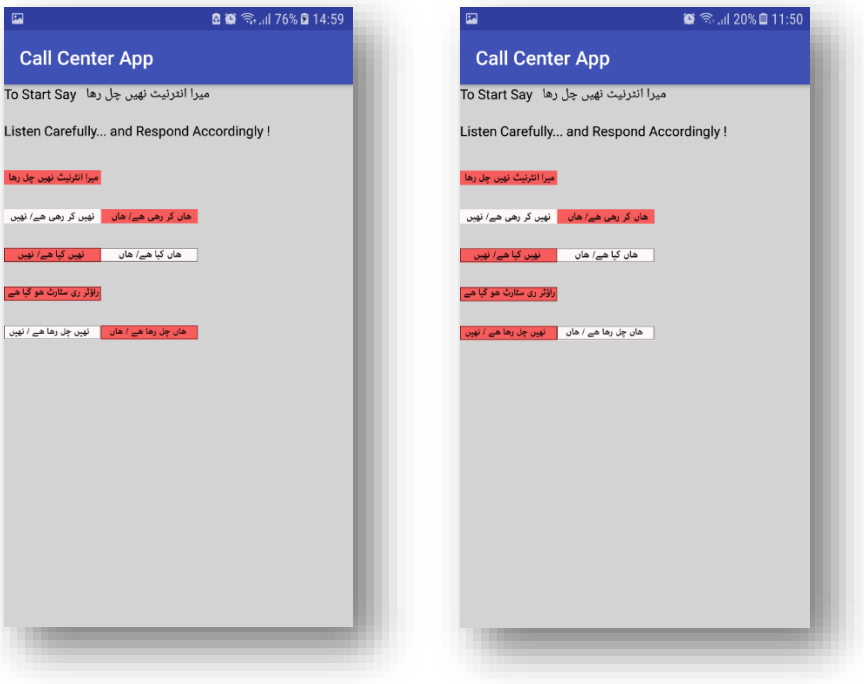


Figure 6.12: Communication Recovered

Figure 6.13: Complaint registered

Figure 6.12 is ending step of the whole process where either user's internet problem will be fixed or his complaint will be registered and he will be given complaint ID. This image shows the red background of "ہاں چل رہا ہے / ہاں" it means the user has confirmed that internet is working now and communication got continued. In this state, it is the end of the problem and it is solved now. So without human involvement and help of the speech recognition user have recuperated his internet.

Figure 6.13 is alternative to Figure 6.12 while both are at same level. Till this step user has restarted the router now either his internet is working or not. In this image, we can see the red background of "نہیں / نہیں چل رہا ہے". Which means that after restarting the router he is not still able to communicate and internet is still not working.

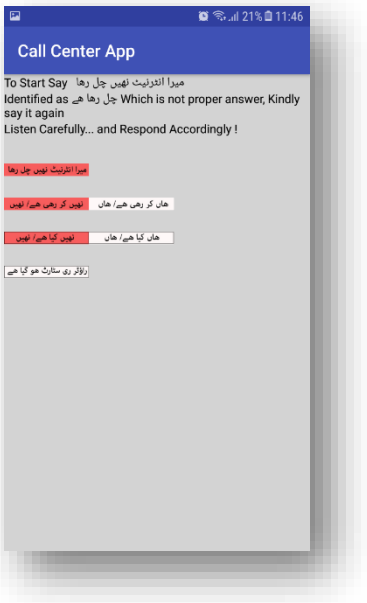


Figure 6.14: unrecognized words

Figure 6.14 is when the user speaks something that system is not expecting. In this case, that line will be shown on screen and user will be requested to speak again. It is valid for many cases like if the user doesn't answer expected an answer, the user speaks in a way that system does not identify it, too much background noise makes it hard to recognize for the system. In all these cases system will say that the identified answer is not an expected answer, kindly try again. So the user has to speak the proper answer in order to move further. This option is valid for all stages.

6.3 Urdu Speech Recognition base Speak and Text:

This application has two main modules. First is command and perform, in this module user give some command by speaking in Urdu and then action related to this command is performed. The second module is speech and write. In this module, the user speaks something and those spoken words are written in the text field. In detail description with screenshots is given below.

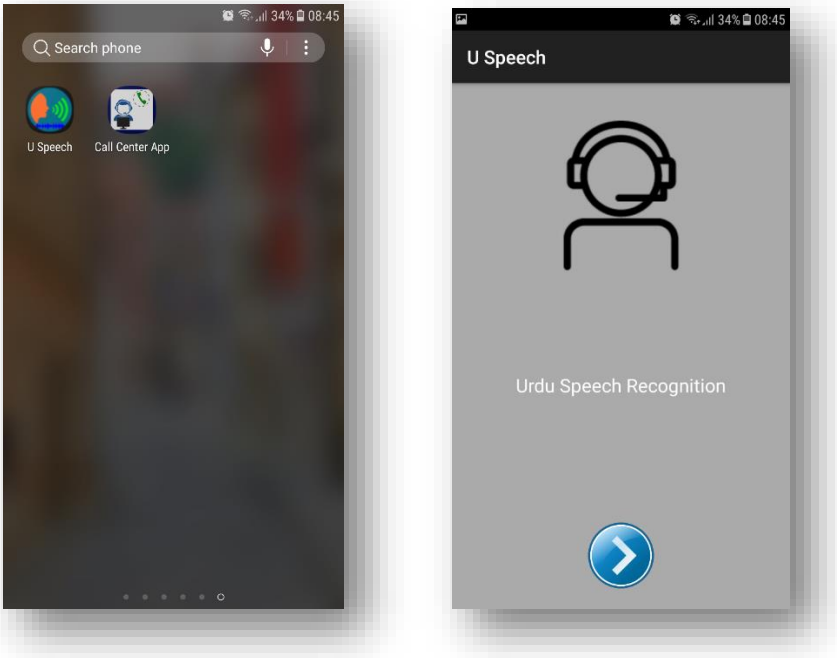


Figure 6.15: Installed application,

Figure 6.16: Home Screen

The application is tested on Samsung Galaxy A520 as OS Android nougat 7.0. Figure 6.15 this is the view of the installed application. Figure 6.16 is home screen of application where the user will click on the arrow button to move on next page.

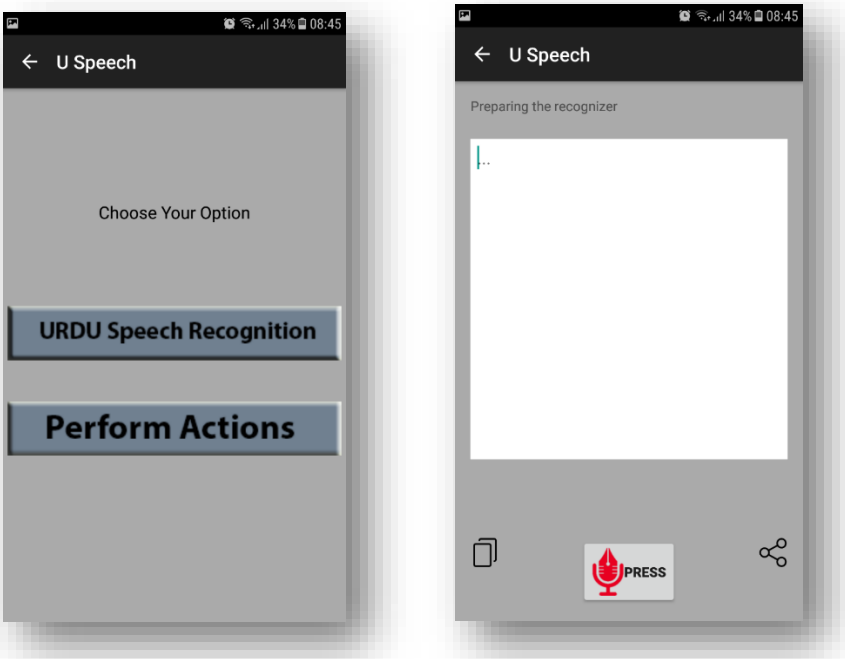


Figure 6.17: Select module

Figure 6.18: USR module

Figure 6.17 is module selection screen, it is the second screen where the user has to select whether he has to go USR module or command module. Here one thing is important if we can notice here is line say preparing the recognizer, in this step we have to wait for the recognizer to be prepared when it will be prepared, this line will be removed. If the line is still written then the button will not be pressed. Furthermore, there is one text filed with white background whatever user will speak and after recognition, it will be written here. Button with mic icon is to speak. Its working defined below. The left side of it, there is copy icon and on right there is share icon.

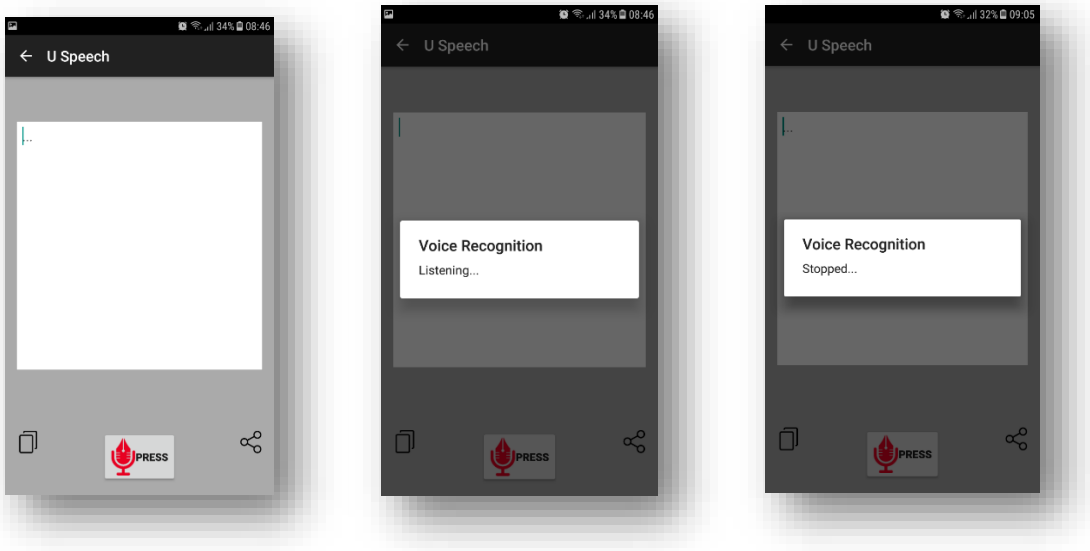


Figure 6.19: Recognizer prepared,

Figure 6.20: Button pressed

Figure 6.21: Button Released.

Figure 6.19 is when recognizer got prepared and is ready to listen. To start speaking user have to keep it pressed. While pressing user will speak and an alert will be shown as in Figure 6.20 and indicate that it is listening. On release of this button, recognition will be stopped and alert value will be changed to stop. Which is indicated in Figure 6.21.

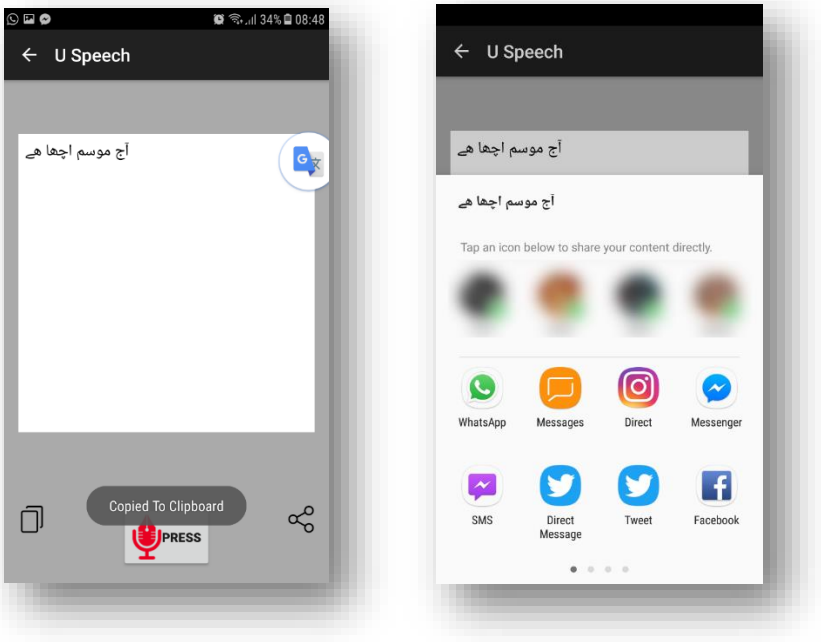


Figure 6.22: Copy Button Pressed

Figure 6.23: Share button pressed.

When copy button will be pressed a toast will be shown that it is copied to the clipboard. Then you can paste this data anywhere else in mobile it is shown in Figure 6.22. When share button is pressed then all application who support this sharing will be shown as displayed in Figure 6.23.

Figure 6.24: Command base opening applications.

This module Figure 6.24 is alternate to Figure 6.17 where the user selected “perform actions” option. These are 15 application that can be open with saying the line as shown on the screen.

To open the application you have to press mic icon and it will start listening and then the user will speak. Figure 6.25 is a demonstration of this. Where the First screen shows the system is listening and then user said "نمبر ملاو" and screen opened.

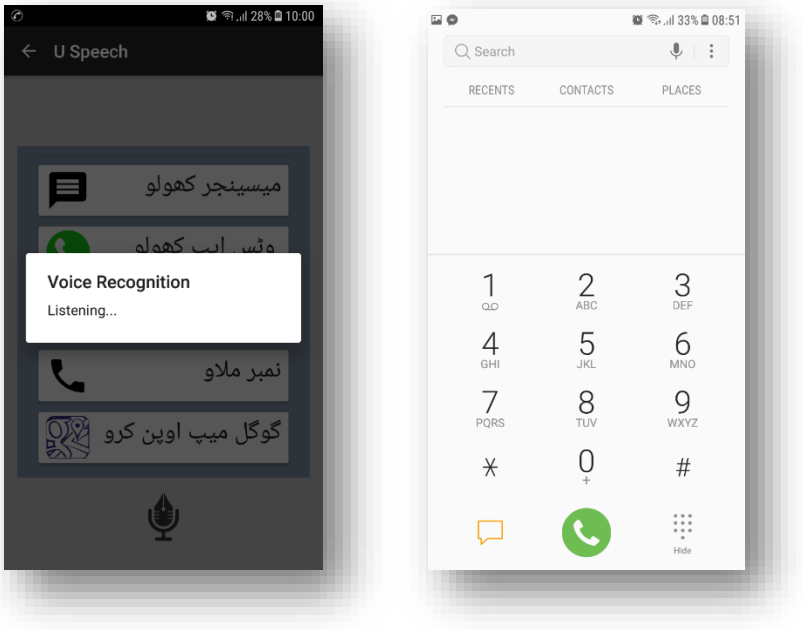


Figure 6.25: System is listening and Dial-Pad is open.

6.4 Conclusion:

This chapter contains a description of both apps made to implement the results of chapter four. The first application is total voice base application where user and system interact with each other. The system makes questions and user response to those questions. This application is helpful for automatic call base application. Where user can make complain about his internet not working. The second application is an android based application where the user can speak and then it will be converted into Urdu text. After this can be shared with other applications or can be copied for further use. Another module of this application is to make commands and perform actions.

Chapter 7

7 Conclusion and Future Work

7.1 Conclusion:

This thesis is titled as “Speech Recognition Based Automated Technical Support System”. The first chapter contains detail information of speech recognition, speech terminologies including definition and examples of the lexicon, phones, language model, word error rate, isolated words, types of utterances, types of speaker models, application of speech. Detail description of Urdu speech and a wide range of speakers of Urdu explaining significance of Urdu ASR. Hidden Markov Model is explained in detail and its usage and importance in speech recognition. In Literature review about 9 years papers are being discussed, Some papers are about corpus creation, some about data collection and recording, WER of papers are being discussed least achieved value is about 60.2% in published work. While in my work least value is 35.7%. Which is best until yet. Also published work maximum size data is 45 hours data while my data consist more than 83 hours data. There are some papers about different numbers of speakers like 65, 81 and 82 but I have collected 181 speakers’ data for just training and 20 speaker’s data for testing. So this data is largest data in Urdu. Chapter three is about methodology and tool, CMUSphinx is a tool used for data training and testing. In this chapter, CMUSphinx is discussed in details, that what are the requirements to use this tool? How these required files and folders can be created? Block diagram of CMUSphinx is discussed in this chapter. Then data collection is explained that how data is collected, from which sources data is collected. How data is cleaned, what types of character causes errors and are needed to be removed. After collecting the written data next step is to record this data. For recording speakers from Sialkot, Gujrat and Gujranwala participated. Recording environment was normal routine life, it was suggested to avoid noise like screaming, and vehicle sounds, the horn of vehicles, and any other person talk in the recording. A tool used for recording is Audacity and default recorder of mobile including Android and iPhone. In the recording, diversity was considered, diversity of speakers, devices, speakers’ age and recording environment. The next step was to train the data in CMUSphinx, This tool uses the specific syntax of data which is defined, Java codes were designed to make these data. Language model was created with different data recording data. Which is made with SRILM toolkit. In first three chapters’ data collection is done. In Chapter four simulation and results are being discussed. To observer error rate in deep detail. Data started from 23.7 hours data training with 55 speakers and testing it with 1 speaker data. In this step, WER was not good so increased data to 48.89 hours and speakers were 105. This data was tested with 5 speakers while in time data was 46 minutes. Then changing the testing data and speakers are done to see the effect on WER. With larger testing data, trained data is checked again and again, their detail is explained in chapter four. Chapter five is about discussion and findings. While multiple time testing and running training too many time I observed multiple error reasons. So I made a tool to compare errors, detail of this tool with the screenshot is given. Also, errors are being compared and shown which type of errors causing an increase in the error rate. There are five types of errors, Actual error, these are errors which are real error and cannot be removed, on my personal observation these errors are mostly on starting of recording or end of the recording. Some errors are due to improper spacing. Some words can be written jointly and individually, but for system these become error.

Details in tabular form are given in the chapter. Some words are due to the replaced character. In Urdu some words can be written in multiple ways, words containing ہ are a most famous category here. Then some words are incorrect words which are either incorrect in a testing file or in training file. The last and 6th chapter contains two android based applications, one is for call center support system where the user can file his complaint without human involvement from system side and can solve his internet connectivity problem. The second application is command and control base app where the user can give the command and can open the application on the android phone. Another module is a speech to text, where user speak and text will be written in text area then this text can be copied to the clipboard and can be shared to other applications. This is the summary of whole document and thesis work which describes all work done to fulfill this thesis requirement.

7.2 Future Work:

In this thesis, I created an acoustic model which give pretty good accuracy in published work. But still this accuracy can be increased and WER can be reduced. Chapter five of this thesis defines the ways to reduce WER, by using them one can work in future to get better results. Furthermore, data used in this work is mostly collected from news and social websites but if we include more diverse data like Urdu Novels which have a lot of literature and maybe some words who may be got missed in this works then a better acoustic model can be created ignoring the WER. This corpus is collected from Pakistani websites and recorded from people of Sialkot, Gujranwala, and Gujrat. But we get recordings from all over Pakistan, even include speakers and corpus form other countries like India then more diverse acoustic model will be created. So increasing training data, including diverse corpus, recording from people of different areas may get efficient results and inclusion of new words in the lexicon. There are very few applications available in android store and in the market related to Urdu. I made two applications using this acoustic model while further so many applications can be made, applications which will be beneficial for disable and illiterate people. And eventually for Pakistan.

Thanks to Allah Almighty

8 References

[1] A. A. Raza, A. Habib, J. Ashraf, and M. Javed, "A Review on Urdu Language Parsing," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 4, pp. 93–97, 2017.

[2] D. O'Shaughnessy, "Interacting with computers by voice: Automatic speech recognition and synthesis," *Proc. IEEE*, vol. 91, no. 9, pp. 1272–1305, 2003.

[3] "Ethnologue: Languages of the World." [Online]. Available: https://www.ethnologue.com/. [Accessed: 02-Mar-2017].

[4] "| Ministry of Finance | Government of Pakistan |." [Online]. Available: http://www.finance.gov.pk/. [Accessed: 02-Mar-2017].

[5] U. Singh, V. Goyal, and G. S. Lehal, "Named Entity Recognition System for Urdu," in *COLING*, 2012.

[6] A. R. Ali and M. Ijaz, "Urdu Text Classification," in *Proceedings of the 7th International Conference on Frontiers of Information Technology*, 2009, p. 21:1--21:7.

[7] F. Adeeba and S. Hussain, "Experiences in building the Urdu WordNet," *Asian Lang. Resour. collocated with IJCNLP 2011*, p. 31, 2011.

[8] N. Durrani and S. Hussain, "Urdu Word Segmentation," in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2010, pp. 528–536.

[9] S. M. J. Rizvi, M. Husssain, and N. Qaiser, "Language Oriented Parsing Through Morphologically Closed Word Classes in Urdu," in *Student Conference On Engineering, Sciences and Technology*, 2004, pp. 19–24.

[10] V. Gupta, N. Joshi, and I. Mathur, "Rule Based Stemmer in Urdu," *CoRR*, vol. abs/1310.0, 2013.

[11] M. Ali, S. Khalid, M. H. Saleemi, W. Iqbal, A. Ali, and G. Naqvi, "Article: A Rule based Stemming Method for Multilingual Urdu Text," *Int. J. Comput. Appl.*, vol. 134, no. 8, pp. 10–18, Jan. 2016.

[12] W. Anwar, X. Wang, L. Li, and X. L. Wang, "A Statistical Based Part of Speech Tagger for Urdu Language," in *2007 International Conference on Machine Learning and Cybernetics*, 2007, vol. 6, pp. 3418–3424.

[13] N. Shmyrev, “CMUSphinx Open Source Speech Recognition.” [Online]. Available: https://cmusphinx.github.io/. [Accessed: 03-Jan-2018].

[14] O. Prabhakar and N. Sahu, “A Survey On: Voice Command Recognition Technique,” *Int. J. Adv. Res. …*, vol. 3, no. 5, pp. 576–585, 2013.

[15] “Speech API - Speech Recognition | Google Cloud Platform.” [Online]. Available: https://cloud.google.com/speech/. [Accessed: 02-Jan-2018].

[16] H. Zen, A. Senior, and M. Schuster, “Statistical parametric speech synthesis using deep neural networks,” in *International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 7962–7966.

[17] H. Bourlard and N. Morgan, “Connectionist speech recognition: a hybrid approach,” in *International Journal on Pattern Recognition and Artificial Intelligence*, 1994, vol. 247, no. 4, pp. 647–668.

[18] “Center for Language Engineering.” [Online]. Available: http://www.cle.org.pk/. [Accessed: 02-Jan-2018].

[19] “National Language Promotion Department - ادارہ فروغ قومی زبان.” [Online]. Available: http://nlpd.gov.pk/. [Accessed: 02-Mar-2018].

[20] S. K. Hasnain and M. S. Awan, “Recognizing spoken urdu numbers using fourier descriptor and neural networks with matlab,” in *2nd International Conference on Electrical Engineering, ICEE*, 2008, no. March, pp. 2–7.

[21] H. Sarfraz *et al.*, “Speech Corpus Development for a Speaker Independent Spontaneous Urdu Speech Recognition System,” in *Proceedings of O-COCOSDA 2010*, 2010, pp. 1–6.

[22] A. R. Agha, H. Sarmad, S. Huda, U. Inam, and S. Zahid, “Design and development of phonetically rich Urdu speech corpus,” pp. 1–5.

[23] M. A. M. Abushariah, R. N. Ainon, R. Zainuddin, M. Elshafei, and O. O. Khalifa, “Natural Speaker-Independent Arabic Speech Recognition System Based on Hidden Markov Models Using Sphinx Tools,” 2010, no. May, pp. 11–13.

[24] J. Ashraf, N. Iqbal, N. S. Khattak, and A. M. Zaidi, “Speaker Independent Urdu speech recognition using HMM,” in *2010 The 7th International Conference on Informatics and Systems (INFOS)*, 2010, pp. 1–5.

[25] A. Stolcke, “Srilm — an Extensible Language Modeling Toolkit,” *Interspeech*, vol. 2, no. Denver, Colorado, pp. 901–904, 2002.

[26] A. Raza, S. Hussain, and H. Sarfraz, “An ASR System for Spontaneous Urdu Speech,” in *the Proc. of Oriental …*, 2010, pp. 1–6.

[27] H. Sarfraz *et al.*, “Large vocabulary continuous speech recognition for Urdu,” in *Proceedings of the 8th International Conference on Frontiers of Information Technology - FIT ’10*, 2010, pp. 1–5.

[28] H. Ali, N. Ahmad, K. M. Yahya, and O. Farooq, “A Medium Vocabulary Urdu Isolated Words Balanced Corpus for Automatic Speech Recognition,” in *Proceedings of 4th International Conference on Electronic Computer Technology, ICECT*, 2012, no. Icect, pp. 473–476.

[29] S. Ali, S. Iqbal, and I. Saeed, “Voice controlled Urdu interface using isolated and continuous speech recognizer,” in *2012 15th International Multitopic Conference, INMIC 2012*, 2012, pp. 53–57.

[30] S. Irtza and S. Hussain, “Minimally balanced corpus for speech recognition,” in *2013 1st International Conference on Communications, Signal Processing and Their Applications, ICCSPA 2013*, 2013, pp. 0–5.

[31] S. A. Ali, B. Ashraf, H. A. Owais, and S. Saifuddin, “Speech recognizer for regional languages of Pakistan,” in *2014 International Conference on Robotics and Emerging Allied Technologies in Engineering, iCREATE 2014 - Proceedings*, 2014, pp. 68–72.

[32] H. Ali, A. Jianwei, and K. Iqbal, “Automatic Speech Recognition of Urdu Digits with Optimal Classification Approach,” in *Ijca*, 2015, vol. 118, no. 9, pp. 1–5.

[33] Asadullah, A. Shaukat, H. Ali, and U. Akram, “Automatic Urdu Speech Recognition using Hidden Markov Model,” in *2016 International Conference on Image, Vision and Computing (ICIVC)*, 2016, pp. 135–139.

[34] M. Qasim, S. Rauf, S. Hussain, and T. Habib, “Urdu speech corpus for travel domain,” in *2016 Conference of The Oriental Chapter of International Committee for Coordination and Standardization of Speech Databases and Assessment Techniques (O-COCOSDA)*, 2016, pp. 237–240.

[35] M. Qasim, S. Nawaz, S. Hussain, and T. Habib, “Urdu speech recognition system for district names of Pakistan: Development, challenges and solutions,” in *2016 Conference of The Oriental Chapter of International Committee for Coordination and Standardization of Speech Databases and Assessment Techniques (O-COCOSDA)*, 2016, pp. 28–32.

[36] C. Madhu, A. George, and L. Mary, “Automatic language identification for seven Indian languages using higher level features,” in *2017 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES)*, 2017, pp. 1–6.

[37] S. Toshniwal *et al.*, “Multilingual Speech Recognition With A Single End-To-End Model,” 2017.

[38] S. Member and J. H. L. Hansen, “Curriculum Learning Based Approaches for Noise Robust Speaker Recognition,” 2018, vol. 26, no. 1, pp. 197–210.

[39] Y. Song, I. Mcloughlin, and L. Dai, “LID-Senones and Their Statistics for,” 2018, vol. 26, no. 1, pp. 171–183.

[40] S. Awais, “Opinion within Opinion: Segmentation Approach for Urdu Sentiment, in-press,” in *The International Arab Journal of Information Technology*, 2016, vol. 15, no. 1, pp. 21–28.

[41] A. Misra and J. H. L. Hansen, “Modelling and compensation for language mismatch in speaker verification,” in *Speech Communication*, 2018, vol. 96, no. November 2017, pp. 58–66.

[42] C. Gaida, P. Lange, R. Petrick, P. Proba, and D. Suendermann-oeft, “Comparing Open-Source Speech Recognition Toolkits,” in *DHBW Stuttgart Technical Report*, 2014.

[43] A. Beg and S. K. Hasnain, “A Speech Recognition System for Urdu Language,” in *Wireless Networks, Information Processing and Systems: International Multi Topic Conference, IMTIC 2008 Jamshoro, Pakistan, April 11-12, 2008 Revised Selected Papers*, D. M. A. Hussain, A. Q. K. Rajput, B. S. Chowdhry, and Q. Gee, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 118–126.

[44] D. Becker and K. Riaz, “A Study in Urdu Corpus Construction,” in *Proceedings of the 3rd Workshop on Asian Language Resources and International Standardization - Volume 12*, 2002, pp. 1–5.

[45] B. Baker, Paul and Hardie, Andrew and McEnery, Tony and Jayaram, “Corpus data for South Asian language processing,” in *Proceedings of the 10th Annual Workshop for South Asian Language Processing, EACL*, 2003.

[46] “Daily Jang Urdu News | Pakistan News | Latest News - Breaking News.” [Online]. Available: https://jang.com.pk/. [Accessed: 02-Mar-2017].

[47] “Daily Express Urdu Newspaper | Latest Pakistan News | Breaking

News.” [Online]. Available: https://www.express.com.pk/. [Accessed: 02-Mar-2017].

[48] “Daily Dunya ePaper | Urdu Newspaper | Pakistan News | City News | Daily Urdu News.” [Online]. Available: http://e.dunya.com.pk/splash.php. [Accessed: 02-Mar-2017].

[49] “Khabrain Epaper.” [Online]. Available: http://epaper.dailykhabrain.com.pk/. [Accessed: 02-Mar-2017].

[50] “Nawaiwaqt - Daily Urdu ePaper - Lahore, Islamabad, Karachi and Multan editions.” [Online]. Available: http://www.nawaiwaqt.com.pk/. [Accessed: 02-Mar-2017].

[51] “آج کا اخبار - Daily Din News ePaper, Pakistani Urdu Newspaper .” [Online]. Available: http://e.dailydinnews.com/home?pg. [Accessed: 02-Mar-2017].

[52] “Audacity® | Free, open source, cross-platform audio software for multi-track recording and editing.” [Online]. Available: http://www.audacityteam.org/. [Accessed: 02-Mar-2017].

[53] “Speech Assistant AAC - Android Apps on Google Play.” [Online]. Available: https://play.google.com/store/apps/details?id=nl.asoft.speechassistant&hl=en. [Accessed: 31-Jan-2018].

[54] “Speechnotes - Speech To Text - Android Apps on Google Play.” [Online]. Available: https://play.google.com/store/apps/details?id=co.speechnotes.speechnotes&hl=en. [Accessed: 31-Jan-2018].

[55] “Speech To Text - Android Apps on Google Play.” [Online]. Available: https://play.google.com/store/apps/details?id=appinventor.ai_xenom_apps.SpeechToText&hl=en. [Accessed: 31-Jan-2018].

[56] “Speech to Text Translator TTS - Android Apps on Google Play.” [Online]. Available: https://play.google.com/store/apps/details?id=com.fsm.speech2text. [Accessed: 31-Jan-2018].

[57] “Text and Drive - Android Apps on Google Play.” [Online]. Available: https://play.google.com/store/apps/details?id=appinventor.ai_xenom_apps.DriveAndText. [Accessed: 31-Jan-2018].