VOICE BASED EMAIL SYSTEM FOR VISUALLY IMPAIRED

A Report for the Evaluation 3 of Project 2

Submitted by
Yashu Chauhan(1613101861)

in partial fulfilment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

Under the Supervision of
Mr. Prabhat Chandra Gupta
Assistant Professor

APRIL / MAY- 2020

SCHOOL OF COMPUTING AND SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

Certified that this project report **"SOME PERFORMANCE ASPECTS CONSIDERATIONS OF A CLASS OF ARTIFICIAL NEURAL NETWORK"** is the bonafide work of **"YASHU CHAUHAN(1613101861)"** who carried out the project work under my supervision.

SIGNATURE OF HEAD

Dr. MUNISH SHABARWAL,
PhD (Management), PhD (CS)
Professor & Dean,
School of Computing Science &
Engineering

SIGNATURE OF SUPERVISOR

Mr. PRABHAT CHANDRA GUPTA
Assistant Professor
School of Computing Science &
Engineering

ABSTRACT

Internet is one of the basic luxury for daily living. Every person is using the facts and information on internet. On the other hand, blind people face difficulty in accessing the text resources. The advancement in computer based accessible systems has opened up many avenues for the visually impaired across a wide majority of the globe. Audio feedback based virtual environment like, the screen readers have helped blind people to access internet applications immensely. However the visually challenged people find it very difficult to utilize this technology because of the fact that using them requires visual perception. Even though many new advancements have been implemented to help them use the computers efficiently no naïve user who is visually challenged can use this technology as efficiently as a normal naïve user can do that is unlike normal users they require some practice for using the available technologies. In this project, the voicemail system architecture that can be used by a blind person to access e-mails easily and efficiently. The contribution made by project has enabled the blind people to send and receive voice-based e-mail message. The proposed system GUI has been evaluated against the GUI of a traditional mail server and found that the proposed architecture performs much better than that of the existing GUIS. In this project, the use of voice to text and text to voice technique access for blind people. Also this system can be used by any normal person also for example the one who is not able to read. The system is completely based on interactive voice response which will make it efficient.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVATIONS

ABBREVATIONS	MEANING
IVR	Interactive Voice Response
DTMF	Dual Tone Multi Frequency
ASR	Automatic Speech Recognition
STT	Speech To Text
TTS	Text To Speech
GUI	Graphical User Interface
PCA	Principle Component Analysis
HTML	Hypertext Markup Language
VRU	Voice Reaction Unit
ADC	Analog To Digital Converter
API	Application Program Interface
gTTS	Google Text To Speech
SMTP	Simple Mail Transfer Protocol
IMAP	Internet Message Access Protocol
TCP	Transmission Control Protocol

Chapter 1
<u>INTRODUCTION</u>

We have seen that the introduction of Internet has revolutionized many fields. Internet has made life of people so easy that people today have access to any information they want easily. Communication is one of the main fields highly changed by Internet.

E-mails are the most dependable way of communication over Internet, for sending and receiving some important information. But there is a certain norm for humans to access the Internet and the norm is you must be able to see. But there are also differently able people in our society who are not gifted with what you have. There are some visually impaired people or blind people who can't see things and thus can't see the computer screen or keyboard.

A survey has shown that there are more than 240 million visually impaired people around the globe. That is, around 240 million people are unaware of how to use Internet or E-mail. The only way by which a visually challenged person can send an E-mail is, they have to speak the entire content of the mail to another person (not visually challenged) and then that third person will compose the mail and send on the behalf of the visually challenged person. But this is not a right way to deal with the problem. It is very unlikely that every time a visually impaired person can find someone for help. Although for these reasons the visually impaired people are criticized by our society.

So, for the betterment of society and giving an equal status to such specially able people we have come up with this project idea which provides the user with ability to send mails using voice commands without the need of keyboard or any other visual things.

1.1 INTERACTIVE VOICE RESPONSE

Interactive voice response (IVR) is a technology that allows a computer to interact with humans through the use of voice and DTMF tones input via a keypad. In telecommunications, IVR allows customers to interact with a company's host system via

a telephone keypad or by speech recognition, after which services can be inquired about through the IVR dialogue. IVR systems can respond with pre-recorded or dynamically generated audio to further direct users on how to proceed.

1.2 SPEECH RECOGNITION

Speech recognition is the inter-disciplinary sub-field of computational linguistics that develops methodologies and technologies that enables the recognition and translation of spoken language into text by computers. It is also known as "automatic speech recognition" (ASR), "computer speech recognition", or just "speech to text" (STT).

1.3 SPEECH TO TEXT

The system acquires speech at run time through a microphone and processes the sampled speech to recognize the uttered text. The recognized text can be stored in a file. We are developing this on Android platform using Eclipse workbench. Our speech to-text system directly acquires and converts speech to text. It can supplement other larger systems, giving users a different choice for data entry. A speech-to-text system can also improve system accessibility by providing data entry options for blind, deaf, or physically handicapped users. Speech recognition system can be divided into several blocks: feature extraction, acoustic models database which is built based on the training data, dictionary, language model and the speech recognition algorithm. Analog speech signal must first be sampled at time and amplitude axes, or digitized. Samples of the speech signal are analyzed in even intervals.This period is usually 20 ms because the signal in this interval is considered stationary. Speech feature extraction involves the formation of equally spaced discrete vectors of speech characteristics. Feature vectors from training database are used to estimate the parameters of acoustic models. The acoustic model describes properties of the basic elements that can be recognized. The basic element can be a phoneme for continuous speech or word for isolated words recognition.

1.4 TEXT TO SPEECH

Converting text to voice output using speech synthesis techniques. Although initially used by the blind to listen to written material, it is now used extensively to convey financial data, e-mail messages, and other information via telephone for everyone. Text-to-speech is also used on handheld devices such as portable GPS units to announce street names when giving directions. Our Text-to-Speech Converter‖ accepts a string of 50 characters of text (alphabets and/or numbers) as input. In this, we have interfaced the keyboard with the controller and defined all the alphabets as well as digits keys on it. The speech processor has an unlimited dictionary and can speak out almost any text provided at the input most of the times. Hence, it has an accuracy of above 90%. It is a microcontroller based hardware coded in Embedded C language. Further research is to be done to optimize various methods of inputting the text i.e. Reading the text using optical sensor and converting it to speech so that almost all sorts of physical challenges faced by the people while communicating are overcome.

1.5 PURPOSE OF THE PROJECT

This project proposes a python based application, designed specifically for visually impaired people. This application provide a voice based mailing service where they could read and send mail on their own, without any guidance through their Email accounts. The VMAIL system can be used by a blind person to access mails easily and adeptly. Hence dependence of visually challenged on other individual for their activities associated to mail can be condensed.

The application will be a python-based application for visually challenged persons using IVR- Interactive voice response, thus sanctioning everyone to control their mail accounts using their voice only and to be able to read, send, and perform all the other useful tasks. The system will ask the user with voice commands to perform certain action and the user will respond to it.

The main advantage of this system is that use of keyboard is completely eliminated, the

user will have to respond through voice only.

1.6 MOTIVATION

It is estimated that nearly 285 million people in the world are visually impaired and idea is to facilitate suitable communication system for them. This reason was driving force behind developing given system. One of the major disadvantages of existing system is that all operations are based on mouse click events and keyboard. Operations depend completely on types of clicks specified by idea. Also sometimes remembering keyboard shortcut is difficult. The extent of existing system is limited for blind and visually impaired people. There is high need of developing a proper system which curbs all the above drawbacks and turn into a simple system. Idea focuses on providing basic functionalities like compose, send, receive E-mail along with advance features like voice based operation, search mail, provision for voice as well as text based email with added ease and simplicity. Related Work Interaction of the users to the system earlier was based on Screen reader based technology and also system based on mouse click based operations were in for every operation there is associated mouse click for example to compose email let say to left clicks. Therefore interaction with the system is tough also there is need to keep events in mind. This paper focuses on developing an email system which helps blind people to use communication services. The system based in IVR is used, major idea is to discard keyboard and use of mouse operation. Internet is rich source of knowledge and information, blind people face difficulties in accessing text based material. The idea is to develop audio feedback based virtual environment like screen reader, text to speech, etc. Voice mail architecture helps blind people to access info. in form of audio, text, self read system. Idea focuses on helping visually impaired and illiterate people to access technology by reducing cognitive load. Decision making depends on eyesight and everything that happens or appears.

Chapter 2

LITERATURE REVIEW

2.1 "Voice Based System in Desktop and Mobile Devices for Blind People". In International Journal of Emerging Technology and Advanced Engineering (IJETAE), 2014

This paper deals with "Voice Based System in Desktop and Mobile Devices for Blind People". Voice mail architecture helps blind people to access e-mail and other multimedia functions of operating system (songs, text).Also in mobile application SMS can be read by system itself. Now a days the advancement made in computer technology opened platforms for visually impaired people across the world. It has been observed that nearly about 60% of total blind population across the world is present in INDIA. In this paper, we describe the voice mail architecture used by blind people to access E-mail and multimedia functions of operating system easily and efficiently. This architecture will also reduce cognitive load taken by blind to remember and type characters using keyboard. There is bulk of information available on technological advances for visually impaired people. This includes development of text to Braille systems, screen magnifiers and screen readers. Recently, attempts have been made in order to develop tools and technologies to help Blind people to access internet technologies. Among the early attempts, voice input and input for surfing was adopted for the Blind people. In IBM's Home page the web page is an easy-to-use interface and converts the text-to-speech having different gender voices for reading texts and links. However, the disadvantage of this is that the developer has to design a complex new interface for the complex graphical web pages to be browsed and for the screen reader to recognize.

Simple browsing solution, which divides a web page into two dimensions. This greatly simplifies a web page's structure and makes it easier to browse. Another web browser

generated a tree structure from the HTML document through analyzing links. As it attempted to structure the pages that are linked together to enhance navigability, it did not prove very efficient for surfing. After, it did not handle needs regarding navigability and usability of current page itself. Another browser developed for the visually handicapped people was eGuideDog which had an integrated TTS engine. This system applies some advanced text extraction algorithm to represent the page in a user-friendly manner. However, still it did not meet the required standards of commercial use. Considering Indian scenario, ShrutiDrishti and WebBrowser for Blind are the two web browser framework that are used by Blind people to access the internet including the emails. Both the systems are integrated with Indian language ASR and TTS systems. But the available systems are not portable for small devices like mobile phones.

2.2 "Voice Based Search Engine and Web page Reader". In International Journal of Computational Engineering Research (IJCER)

This paper aims to develop a search engine which supports Man-Machine interaction purely in the form of voice. A novel Voice based Search Engine and Web-page Reader which allows the users to command and control the web browser through their voice, is introduced. The existing Search Engines get request from the user in the form of text and respond by retrieving the relevant documents from the server and displays in the form of text .Even though the existing web browsers are capable of playing audios and videos, the user has to request by typing some text in the search text box and then the user can play the interested audio/video with the help of Graphical User Interfaces (GUI). The proposed Voice based Search Engine aspires to serve the users especially the blind in browsing the Internet. The user can speak with the computer and the computer will respond to the user in the form of voice. The computer will assist the user in reading the documents as well. Voice-enabled interface with addition support for gesture based input and output approaches are for the "Social Robot Maggie" converting it into an aloud

reader . This voice recognition and synthesis can be affected by number of reasons such as the voice pitch, its speed, its volume etc. It is based on the Loquendo ETTS (Emotional Text-To-Speech) software. Robot also expresses its mood through gesture that is based on gestionary. Speech recognition accuracy can be improved by removal of noise. In A Bayesian scheme is applied in a wavelet domain to separate the speech and noise components in a proposed iterative speech enhancement algorithm. This proposed method is developed in the wavelet domain to exploit the selected features in the time frequency space representation. It involves two stages: a noise estimate stage and a signal separation stage. In the Principle Component Analysis (PCA) based HMM for the visual modality of audio-visual recordings is used. PCA (Principle Component Analysis) and PDF (Probabilistic Density Analysis). Presents an approach to speech recognition using fuzzy modelling and decision making that ignores noise instead of its detection and removal. In the speech spectrogram is converted into a fuzzy linguistic description and this description is used instead of precise acoustic features.In Voice recognition technique combined with facial feature interaction to assist virtual artist with upper limb disabilities to create visual cut in a digital medium, preserve the individuality and authenticity of the art work. Techniques to recover phenomena such as Sentence Boundaries, Filler words and Disfluencies referred to as structural Metadata are discussed in and describe the approach that automatically adds information about the location of sentence boundaries and speech disfluencies in order to enrich speech recognition output. Clarissa a voice enabled procedure browser that is deployed on the international space station (ISS). The main components of the Clarissa system are speech recognition module a classifier for executing the open microphone accepts/reject decision, a semantic analysis and a dialog manager. Mainly focuses on expressions.

To build a prosody model for each expressive state, an end pitch and a delta pitch for each syllable are predicted from a set of features gathered from the text. The expression-tagged units are then pooled with the neutral data, In a TTS system, such paralinguistic events efficiently provide clues as to the state of a transaction, and Markup specifying

these events is a convenient way for a developer to achieve these types of events in the audio coming from the TTS engine.

Main features of are smooth and natural sounding speech can be synthesized, the voice characteristics can be changed, it is "trainable. Limitations of the basic system is that synthesized speech is "buzz" since it is based on a vocoding technique, it has been overcome by high quality vocoder and hidden semi-Markov model based acoustic modelling. Speech synthesis consists of three categories: Concatenation Synthesis, Articulation Synthesis, and Formant Synthesis.

Mainly focuses on formant synthesis, array of phoneme of syllable with formants frequency is given as input, frequency of given input is processed, on collaborated with Thai-Tonal-Accent Rules convert given formants frequency format to wave format, so that audio output via soundcard.
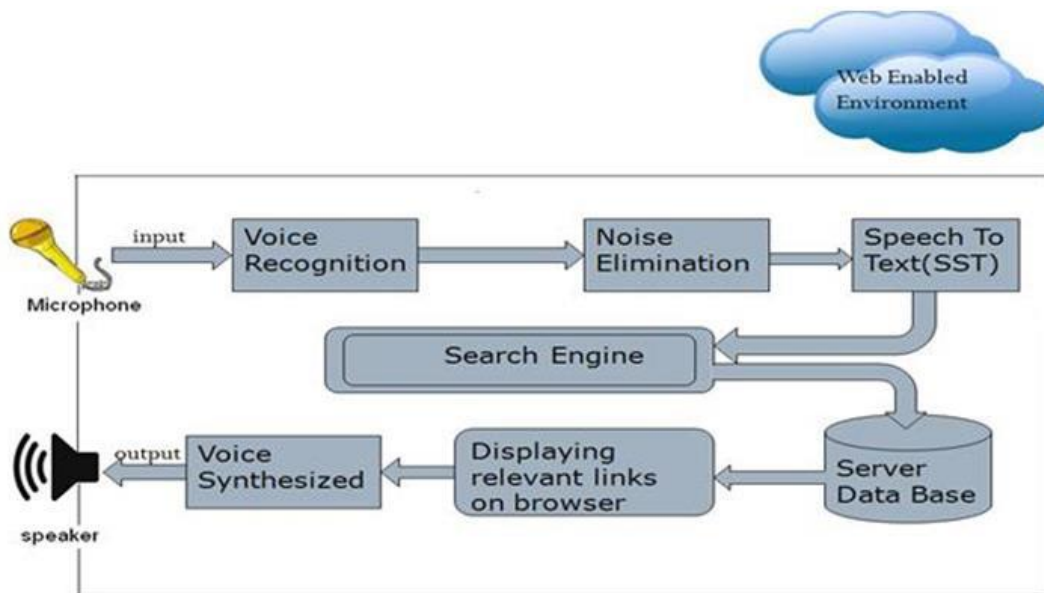
Figure-1: Voice Recognition Flow Diagram

2.3 "Voice Based Services for Blind People". In International Journal of Advance Research, Ideas and Innovations in Technology(IJARIIT)

The advancement in computer based accessible systems has opened up many avenues for the visually impaired across a wide majority of the globe. Audio feedback based virtual environment like, the screen readers have helped blind people to access internet applications immensely. However, a large section of visually impaired people in different countries, in particular, the Indian sub-continent could not benefit much from such systems. This was primarily due to the difference in the technology required for Indian languages compared to those corresponding to other popular languages of the world. In this paper, we describe the voicemail system architecture that can be used by a blind person to access e-mails easily and efficiently. The contribution made by this research has enabled the blind people to send and receive voice-based e-mail messages in their native language with the help of a mobile device. Our proposed system GUI has been evaluated against the GUI of a traditional mail server. We found that our proposed architecture performs much better than that of the existing GUIS. In this project, we use voice to text and text to voice technique access for blind people.

The navigation system uses TTS (Text-to-Speech) for blindness in order to provide a navigation service through voice. Suggested system, as an independent program, is fairly cheap and it is possible to install onto Smartphone held by blind people.This allows blind people to easy access the program. An increasing number of studies have used technology to help blind people to integrate more fully into a global world. We present software to use mobile devices by blind users. The software considers a system of instant messenger to favor interaction of blind users with any other user connected to the network. Nowadays the advancement made in computer technology opened platforms for visually impaired people across the world. It has been observed that nearly about 60% of the total blind population across the world is present in INDIA. In this paper, we describe the voice mail architecture used by blind people to access E-mail and

multimedia functions of the operating system easily and efficiently.This architecture will also reduce cognitive load taken by the blind to remember and type characters using the keyboard. It also helps handicapped and illiterate people. In previous work, blind people does not send email using the system. The multitude of email types along with the ability setting enables their use in nomadic daily contexts. But these emails are not useful in all types of people such as blind people they can't send the email. Audio based email are only preferable for blind peoples. They can easily respond to the audio instructions. In this system is very rare. So there is less chance to available this audio based email to the blind people. We describe the voicemail system architecture that can be used by a blind person to access e-mails easily and efficiently. The contribution made by this research has enabled the blind people to send and receive voice-based e-mail messages in their native language with the help of a computer or a mobile device. Our proposed system GUI has been evaluated against the GUI of a traditional mail server. We found that our proposed architecture performs much better than that of the existing GUIS.

It involves the development of the following modules:

SPEECH_ TO_ TEXT Converter :The system acquires speech at run time through a microphone and processes the sampled speech to recognize the uttered text. The recognized text can be stored in a file. We are developing this on Android platform using Eclipse workbench. Our speech to-text system directly acquires and converts speech to text. It can supplement other larger systems, giving users a different choice for data entry. A speech-to-text system can also improve system accessibility by providing data entry options for blind, deaf, or physically handicapped users. Speech recognition system can be divided into several blocks: feature extraction, acoustic models database which is built based on the training data, dictionary, language model and the speech recognition algorithm. Analog speech signal must first be sampled at time and amplitude axes, or digitized. Samples of the speech signal are analyzed in even intervals. This period is usually 20 ms because the signal in this interval is considered stationary. Speech feature extraction involves the formation of equally spaced discrete vectors of speech

characteristics. Feature vectors from training database are used to estimate the parameters of acoustic models. The acoustic model describes properties of the basic elements that can be recognized. The basic element can be a phoneme for continuous speech or word for isolated

words recognition.

TEXT_ TO_ SPEECH Converter: Converting text to voice output using speech synthesis techniques. Although initially used by the blind to listen to written material, it is now used extensively to convey financial data, e-mail messages, and other information via telephone for everyone. Text-to-speech is also used on handheld devices such as portable GPS units to announce street names when giving directions. Our Text-to-Speech Converter‖ accepts a string of 50 characters of text (alphabets and/or numbers) as input. In this, we have interfaced the keyboard with the controller and defined all the alphabets as well as digits keys on it. The speech processor has an unlimited dictionary and can speak out almost any text provided at the input most of the times. Hence, it has an accuracy of above 90%. It is a microcontroller based hardware coded in Embedded C language. Further research is to be done to optimize various methods of inputting the text i.e. Reading the text using optical sensor and converting it to speech so that almost all sorts of physical challenges faced by the people while communicating are overcome.

WORD RECOGNITION :Voice recognition software (also known as speech to text software)allows an individual to use their voice instead of typing on a keyboard. Voice recognition may be used to dictate text into the computer or to give commands to the computer. Voice recognition software allows for a quick method of writing onto a computer. It is also useful for people with disabilities who find it difficult to use the keyboard. This software can also assist those who have difficulty with transferring ideas onto paper as it helps take the focus out of the mechanics of writing. Word recognition is measured as a matter of speed, such that a word with a high level of recognition is read faster than a novel one. This manner of testing suggests that comprehension

of the meaning of the words being read is not required, but rather the ability to recognize

them in a way that allows proper pronunciation. Therefore, context is unimportant, and word recognition is often assessed with words presented in isolation in formats such as flash cards Nevertheless, ease in word recognition, as in fluency, enables proficiency that fosters comprehension of the
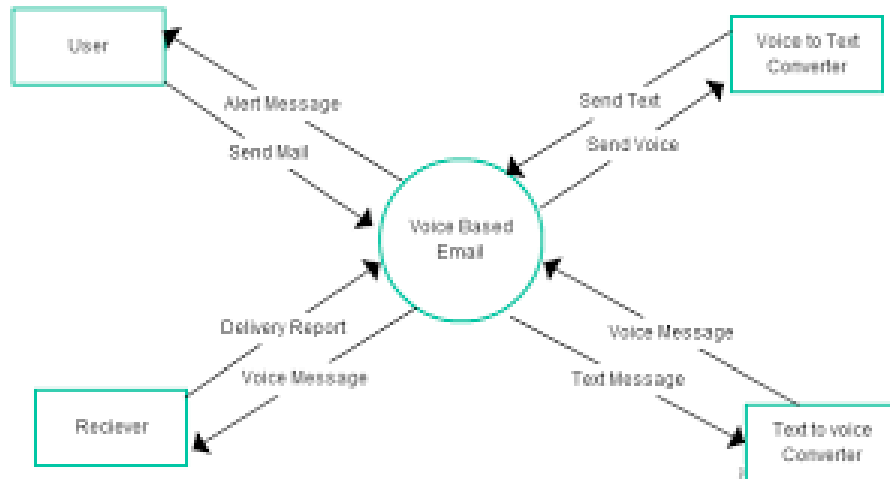
text being read.

Figure-2: System Data Flow Diagram

2.4 "Voice based e-mail System for Blinds". In International Journal of Research Studies in Computer Science and Engineering (IJRSCSE)

Internet plays a vital role in today's world of communication. Today the world is running on the basis of internet. No work can be done without use of internet. Electronic mail i.e. email is the most important part in day to day life. But some of the people in today's world don't know how to make use of internet, some are blind or some are illiterate. So it goes very difficult to them when to live in this world of internet. Nowadays there are various technologies available in this world like screen readers, ASR, TTS, STT, etc. but these are not that much efficient for them. Around 39 million people are blind and 246 people have low vision and also 82 of people living with blindness are 50 aged and above. We have to make some internet facilities to them so

they can use internet. Therefore we came up with our project as voice based email system for blinds which will help a lot to visually impaired peoples and also illiterate peoples for sending their mails. The users of this system don't need to remember any basic information about keyboard shortcuts as well as location of the keys. Simple mouse click operations are needed for functions making system easy to use for user of any age group. Our system provides location of where user is prompting through voice so that user doesn't have to worry about remembering which mouse click operation

The visually challenged people find it very difficult to utilize this technology because of the fact that using them requires visual perception. However not all people can use the internet. This is because in order to access the internet you would need to know what is written on the screen. If that is not visible it is of no use. This makes internet a completely useless technology for the visually impaired and illiterate people.

In this system mainly three types of technologies are used namely:

STT (Speech-to-text): here whatever we speak is converted to text. Their will a small icon ofmic on whose clicking the user had to speak and his/her speech will be converted to text format, which the naked people would see and read also.

TTS (text-to-speech) this, method is full opposite of STT. In this method, which converts the text format of the emails to synthesized speech? A text-to-speech (TTS) system converts language text into speech, alternative systems render symbolic linguistic representations. Synthesized speech can be created by concatenating pieces of recorded speech that are stored in a database.

IVR (Interactive voice response): IVR is an advanced technology describes the interaction between the user and the system in the way of responding by using keyboard for the respective voice message. IVR allows user to interact with an email host system via a system keyboard, after that users can easily service their own enquiries by listening

to the IVR dialogue. IVR systems generally respond with pre-recorded audio voice to further assist users on how to proceed.

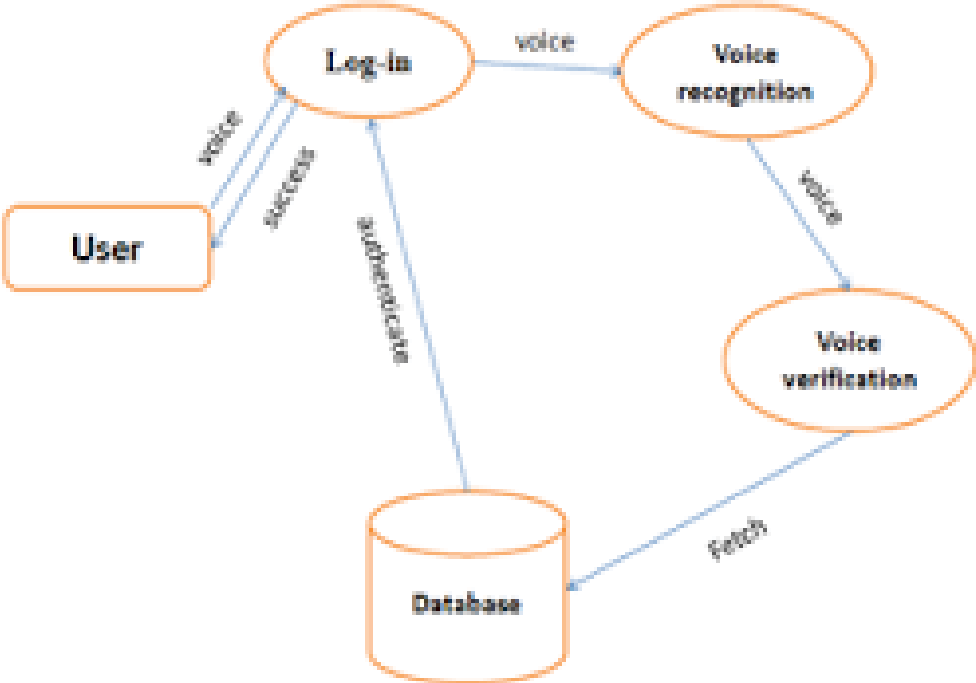The audio that would be pre-recorded and the system need to have large volumes.

Figure-3: Proposed System Architecture

Chapter 3
SYSTEM DEVELOPMENT

3.1 EXISTING SYSTEM

There are a complete number of 4.1 billion email accounts made until 2014 and a there will be evaluated 5.2 billion records by end of 2018. This makes messages the most utilized type of correspondence. The most generally perceived mail benefits that we use in our regular day to day existence can't be used by ostensibly tried people. This is on the grounds that they don't give any office so the individual in front can hear out the substance of the screen. As they can't imagine what is now present on screen they can't make out where to click so as to play out the necessary tasks. For an outwardly tested individual utilizing a PC just because isn't that helpful for what it's worth for an ordinary client despite the fact that it is easy to understand. In spite of the fact that there are many screen readers accessible then likewise these individuals face some minor troubles. Screen readers read out whatever substance is there on the screen and to play out those activities the individual should utilize console alternate routes as mouse area can't be followed by the screen reader. This implies two things; one that the client can't utilize mouse pointer as it is totally awkward if the pointer area can't be followed and second that client ought to be knowledgeable with the console concerning where every single key is found. A client is new to PC can accordingly not utilize this administration as they don't know about the key areas. Another disadvantage that sets in is that screen reader read out the substance in successive way and subsequently client can make out the substance of the screen just on the off chance that they are in essential HTML position. Therefore the new propelled pages which don't follow this worldview so as to make the site more easy to use just make additional issues for these individuals. Moreover the systems that do use only voice for interaction between the user and the system don't have good voice transcription. All these are a few downsides of the present framework which we will defeat in the framework we are creating.

3.2 PROPOSED SYSTEM

The proposed system is based on a completely novel idea and is nowhere like the existing mail systems. The most important aspect that has been kept in mind while developing the proposed system is accessibility. A web system is said to be perfectly accessible only if it can be used efficiently by all types of people whether able or disable. The current systems do not provide this accessibility. Thus the system we are developing is completely different from the current system. Unlike current system which emphasizes more on user friendliness of normal users, our system focuses more on user friendliness of all types of people including normal people visually impaired people as well as illiterate people. The complete system is based on IVR- interactive voice response.

When using this system the computer will be prompting the user to perform specific operations to avail respective services and if the user needs to access the respective services then he/she needs to perform that operation. One of the major advantages of this system is that user won't require to use the keyboard. All operations will be based on voice commands. This system will be perfectly accessible to all types of users as it is just based on simple speech inputs and there is no need to remember keyboard shortcuts. Also because of IVR facility those who cannot read need not worry as they can listen to the prompting done by the system and perform respective actions.

3.3 COMPONENT DESCRIPTION

The proposed system majorly focuses on the use of four main technologies. These technologies can be categorized as the following modules:

3.3.1 DETAIL DESCRIPTION OF IVR

IVR is an advancement that allows a PC to work together with individuals utilizing voice and DTMF tones contribution through a keypad. In media interchanges, IVR licenses customers to connect with an association's host system by methods for a telephone keypad or by talk affirmation, after which organizations can be inquired about through the IVR exchange. IVR systems can respond with pre-recorded or effectively

delivered sound to furthermore control customers on the most capable strategy to proceed. IVR structures sent in the framework are assessed to manage colossal call volumes and besides used for outbound calling, as IVR systems are more wise than various judicious dialer systems. The term voice reaction unit (VRU) is here and there utilized too. IVR systems can be used for convenient purchases, banking portions and organizations, retail orchestrates, utilities, travel information and atmosphere conditions Various advancements fuse using substance to-talk (TTS) to talk staggering and dynamic information, for instance, messages, news reports or atmosphere information. IVR advancement is similarly being brought into vehicle structures for without hands movement. TTS is PC created mixed talk that is never again the robotized voice generally associated with PC. Certified voices make the talk in pieces that are associated and smoothed before being played to the visitor.

The IVR proposes a few benefits that makes it an ideal technology in the development of the project.

Increase first contact resolution:

IVR significantly increases first contact resolution because callers are always directed to the agent who is most capable of meeting their needs or the most appropriate department. The agent who receives the call will be more qualified to answer the caller's question and will be less likely to transfer the call to another agent.

Increase customer service efficiency :

Agents who work in a company that uses an IVR are more proficient at solving specific problems and meeting specific needs of the customers that they are assigned. The result is an increase in customer service efficiency.

Increase agent and company efficiency:

Agents who work in a company with an IVR are more skilled at addressing specific issues, are less likely to consult with colleagues or a manager and are also less likely to

transfer the call to another agent. This results in a significant increase in agent and company efficiency.

Reduce operational costs :

IVR systems will replace a receptionist or a customer service agent who answers calls and directs calls to agents. They are also very affordable, will increase efficiency and will reduce operational costs, so the ROI is huge.

Increase professionalism :

You can use an IVR system to greet your customers in a very professional manner and to make it appear that you have more departments and employees than you actually have.

Increase customer satisfaction:

When your IVR is easy to use and reliable, customers will never be routed to the wrong department, or to an agent who cannot solve their problems.

3.3.2 WORKING OF SPEECH RECOGNITION

Speech recognition is the inter-disciplinary sub-field of computational linguistics that develops methodologies and technologies that enables the recognition and translation of spoken language into text by computers. It is also known as "automatic speech recognition" (ASR), "computer speech recognition", or just "speech to text" (STT). It incorporates knowledge and research in the linguistics, computer science, and electrical engineering fields.Some speech recognition systems require "training" (also called "enrollment") where an individual speaker reads text or isolated vocabulary into the system. The system analyzes the person's specific voice and uses it to fine-tune the recognition of that person's speech, resulting in increased accuracy. Systems that do not use training are called "speaker independent" systems. Systems that use training are called "speaker dependent".

Speech recognition applications include voice user interfaces such as voice dialing (e.g.

"Call home"), call routing (e.g. "I would like to make a collect call"), domotic appliance control, search (e.g. find a podcast where particular words were spoken), simple data entry (e.g., entering a credit card number), preparation of structured documents (e.g. a radiology report), speech-to-text processing (e.g., word processors or emails), and aircraft (usually termed Direct Voice Input).

The term voice recognition or speaker identification refers to identifying the speaker, rather than what they are saying. Recognizing the speaker can simplify the task of translating speech in systems that have been trained on a specific person's voice or it can be used to authenticate or verify the identity of a speaker as part of a security process.

From the technology perspective, speech recognition has a long history with several waves of major innovations. Most recently, the field has benefited from advances in deep learning and big data. The advances are evidenced not only by the surge of academic papers published in the field, but more importantly by the worldwide industry adoption of a variety of deep learning methods in designing and deploying speech recognition systems. Speech recognition works using algorithms through acoustic and language modeling. Acoustic modeling represents the relationship between linguistic units of speech and audio signals; language modeling matches sounds with word sequences to help distinguish between words that sound similar. Often, hidden Markov models are used as well to recognize temporal patterns in speech to improve accuracy within the system. The most frequent applications of speech recognition within the enterprise include call routing, speech-to-text processing, voice dialing and voice search. While convenient, speech recognition technology still has a few issues to work through, as it is continuously developed. The pros of speech recognition software are it is easy to use and readily available. Speech recognition software is now frequently installed in computers and mobile devices, allowing for easy access. The downside of speech recognition includes its inability to capture words due to variations of pronunciation, its lack of support for most languages outside of English and its inability to sort through background noise. These factors can lead to inaccuracies.

Speech recognition performance is measured by accuracy and speed. Accuracy is measured with word error rate. WER works at the word level and identifies inaccuracies in transcription, although it cannot identify how the error occurred. Speed is measured with the real-time factor. A variety of factors can affect computer speech recognition performance, including pronunciation, accent, pitch, volume and background noise.It is important to note the terms speech recognition and voice recognition are sometimes used interchangeably. However, the two terms mean different things. Speech recognition is used to identify words in spoken language. Voice recognition is a biometric technology used to identify a particular individual's voice or for speaker identification.

3.3.3 SPEECH TO TEXT CONVERTER

The process of converting spoken speech or audio into text is called speech to text converter. The process is usually called speech recognition. The Speech recognition is used to characterize the broader operation of deriving content from speech which is known as speech understanding. We often associate the process of identifying a person from their voice, which is voice recognition or speaker recognition so it is wrong to use this term for it.

As shown in the above block diagram speech to text converters depends mostly on two models 1.Acoustic model and 2.Language model. Systems generally use the pronunciation model. It is really imperative to learn that there is nothing like a universal speech recognizer. If you want to get the best quality of transcription, you can specialize the above models for the any given language communication channel.

Likewise another pattern recognition technology, speech recognition can also not be without error. Accuracy of speech transcript deeply relies on the voice of the speaker , the characteristic of speech and the environmental conditions. Speech recognition is a tougher method than what folks unremarkably assume, for a personality's being. Humans are born for understanding speech, not to transcribing it, and solely speech that's well developed will be transcribed unequivocally. From the user's purpose of read,

a speech to text system will be categorized based in its use.

To convert speech to on-screen text or a computer command, a computer has to go through several complex steps. When you speak, you create vibrations in the air. The analog-to-digital converter (ADC) translates this analog wave into digital data that the computer can understand. To do this, it samples, or digitizes, the sound by taking precise measurements of the wave at frequent intervals. The system filters the digitized sound to remove unwanted noise, and sometimes to separate it into different bands of frequency (frequency is the wavelength of the sound waves, heard by humans as differences in pitch). It also normalizes the sound, or adjusts it to a constant volume level. It may also have to be temporally aligned. People don't always speak at the same speed, so the sound must be adjusted to match the speed of the template sound samples already stored in the system's memory.

Next the signal is divided into small segments as short as a few hundredths of a second, or even thousandths in the case of plosive consonant sounds -- consonant stops produced by obstructing airflow in the vocal tract -- like "p" or "t." The program then matches these segments to known phonemes in the appropriate language. A phoneme is the smallest element of a language -- a representation of the sounds we make and put together to form meaningful expressions. There are roughly 40 phonemes in the English language (different linguists have different opinions on the exact number), while other languages have more or fewer phonemes.

The next step seems simple, but it is actually the most difficult to accomplish and is the is focus of most speech recognition research. The program examines phonemes in the context of the other phonemes around them. It runs the contextual phoneme plot through a complex statistical model and compares them to a large library of known words, phrases and sentences. The program then determines what the user was probably saying and either outputs it as text or issues a computer command.
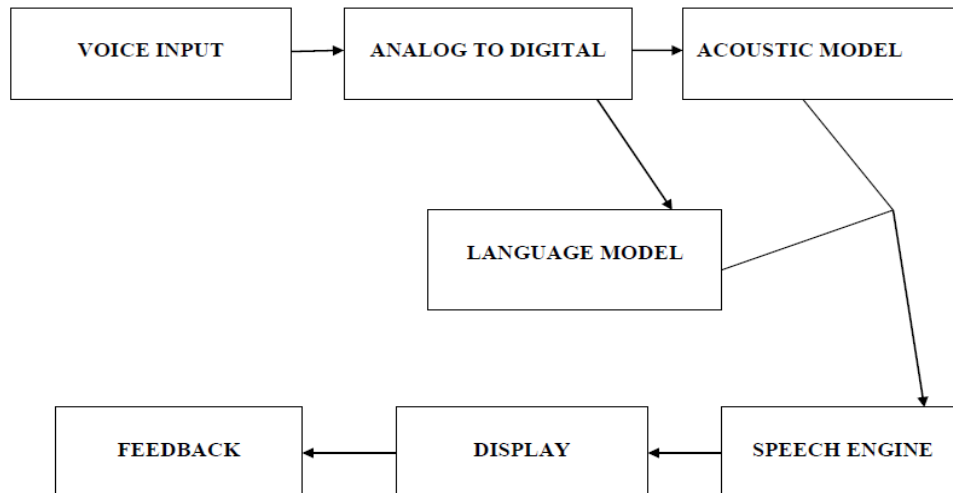
Figure-4: System Block Diagram for Speech Recognition

3.3.4 SPEECH SYSTHESIS

Speech synthesis is the synthetic production of speech. An automatic data handing out system used for this purpose is called as speech synthesizer, and may be enforced in software package and hardware product. A text-to-speech (TTS) system converts language text into speech, alternative systems render symbolic linguistic representations. Synthesized speech can be created by concatenating pieces of recorded speech that are stored in a database. Systems differ in the size of the stored speech units; a system that stores phones or diaphones provides the largest output range, but may lack clarity. For specific usage domains, the storage of entire words or sentences allows for high-quality output. Alternatively, a synthesizer can incorporate a model of the vocal tract and other human voice characteristics to create a completely "synthetic" voice output.

The quality of a speech synthesizer is judged by its similarity to the human voice and by its ability to be understood clearly. An intelligible text to speech program permits individual with ocular wreckage or reading disabilities to concentrate to written words on a computing device. Several computer operational systems have enclosed speech

synthesizers since the first nineteen nineties years.

The text to speech system is consisting of 2 parts:-front-end and a back-end. The front-end consist of 2 major tasks. Firstly, it disciple unprocessed text containing symbols like numbers and abstraction into the equivalent of written out words. This method is commonly known as text, standardization, or processing. Front end then assigns spoken transcriptions to every word, and divides and marks the text into speech units, like phrases, clauses, and sentences.

The process of assigning phonetic transcriptions to words is called text-to-phoneme or grapheme-to-phoneme conversion. Phonetic transcriptions and prosody information together make up the symbolic linguistic representation that is output by the front-end. The back-end—often referred to as the synthesizer—then converts the symbolic linguistic representation into sound. In certain systems, this part includes the computation of the target prosody (pitch contour, phoneme durations), which is then imposed on the output speech.
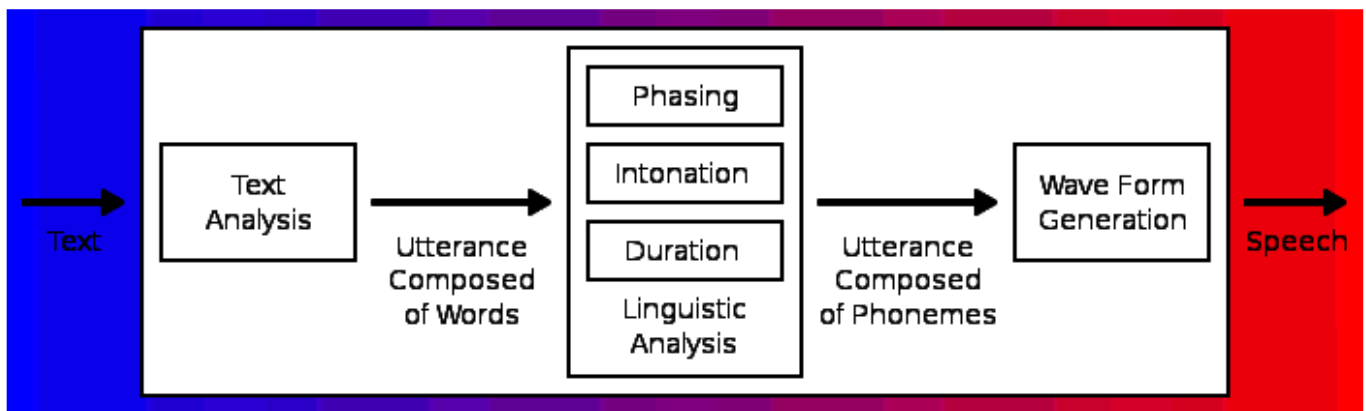
Figure-5: Text to Speech Conversion

Text-to-speech (TTS) is a type of speech synthesis application that is used to create a spoken sound version of the text in a computer document, such as a help file or a Web page. TTS can enable the reading of computer display information for the visually challenged person, or may simply be used to augment the reading of a text message.

Current TTS applications include voice-enabled e-mail and spoken prompts in voice response systems. TTS is often used with voice recognition programs. There are numerous TTS products available, including Read Please 2000, Proverbe Speech Unit, and Next Up Technology's TextAloud. Lucent, Elan, and AT&T each have products called "Text-to-Speech".

In addition to TTS software, a number of vendors offer products involving hardware, including the Quick Link Pen from WizCom Technologies, a pen-shaped device that can scan and read words; the Road Runner from Ostrich Software, a handheld device that reads ASCII text; and DecTalk TTS from Digital Equipment, an external hardware device that substitutes for a sound card and which includes an internal software device that works in conjunction with the PC's own sound card.

Chapter 4

DESIGN

4.1 Design Phases of The Proposed System

A. Phase-1:

The tasks that can be performed using the program developed will be prompted using the voice prompt. In background python module pyttsx3 is used for text to speech conversion.

User will be asked to provide input for the following tasks written below.

 The input is expected in the form of speech by the user which will be converted to text by the Google speech application interface in python and accordingly tasks will be performed.

• Login to their Gmail account.

• Send e-mail through Gmail.

• Read e-mail through Gmail.

B. Phase-2:

In phase-2 of our program the user will give speech input to the system.

This speech input will be handled by speech_recognition module.It is a python library which is used to handle the voice requests and it converts speech into text.

Now after receiving input from the user speech to text converter will save the response in respective variables used in the script and based on their value it will further enter into respective modules.

C. Phase-3:

In this phase our program will handle the requests by the user. Based on the speech input given by the user it will launch the modules.

• Login to G-mail account:- This module will handle the request by user to login in their

g-mail account. This module will make the connection with the user's gmail account based on the credentials provided through voice input. This module's script designed as such it will prompt user to enter their g-mail username and password and then it will use selenium web-driver to automate the task for the user and as a result connection will be made.

• Send E-mail through G-mail:- This module will handle the request by user to send email through their g-mail account. The python script for this module will prompt the user to enter their credentials and then it will make connection with their account.
After the connection has been done it will further prompt the user to enter the receiver's account e- mail id and it will then allow the user to speak their message and it will repeat it for them and by saying ok it will send the mail.
SMTP library in python is used for the above task.

• Read E-mail through G-mail:- This module will handle the request by user to read email through their g-mail account. The python script for this module will prompt the user to enter their credentials and then it will make connection with their account.
After the connection has been done it will start fetching the unread mails for the user and will speak it for them with the help of pyttsx3 or gTTS library in python for text to speech conversion.
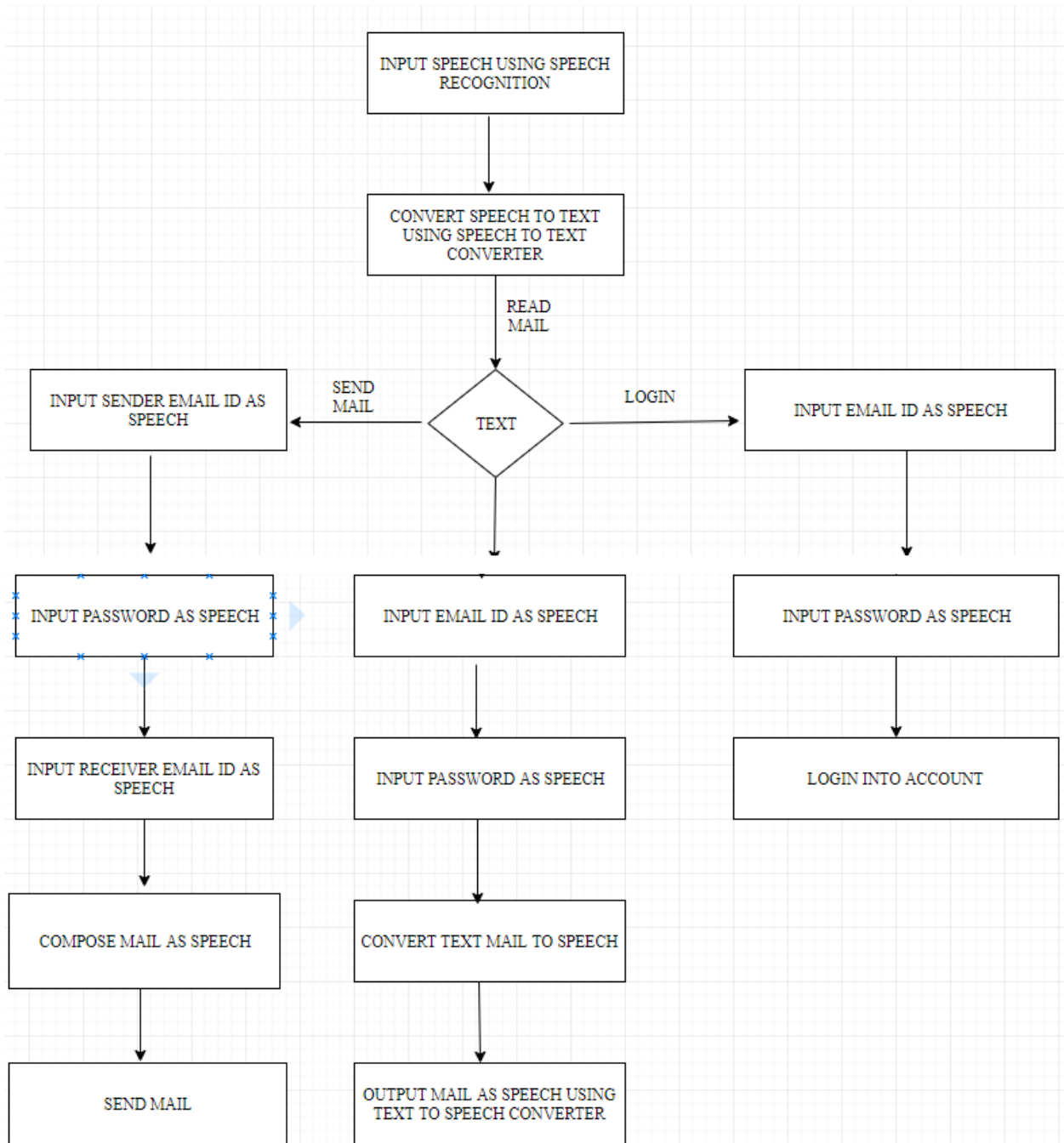
Figure-6: Illustration of Sending and Receiving E-mails

Chapter 5

IMPLEMENTATION

5.1 SPEECH RECOGNITION IN PYTHON

The improvement and accessibility alone in the field of speech recognition are worth considerable. It allows the physically and the elderly and visually challenged people to collaborate with state of the art products and services quickly and naturally no graphical user interface is needed.

If you want to use speech recognition or simply convert speech to text in your python it is very easy to use. Let's see how:-

• Working of speech recognition.

• Packages available in PyPI.

• How to install and how to use speech recognition package using python library.

A handful of packages for speech recognition exist on PyPI. A few of them include:

• Google-cloud-speech

• Watson-developer-cloud

• Pocketsphinx

• Wit

• Apiai

• Speech Recognition

SpeechRecognition is a library that acts as a wrapper for many popular speech APIs and is thus very flexible to use. One of these is the Google Web Speech API which supports a default API key that is hard coded into the SpeechRecognition library.

The elasticity and easy to use features of the SpeechRecognition package in python make it a very good choice for developers who are working on any python project. It does not guarantee to support every feature that is wrapped with this API. You will have

to dispense some time searching for the easily available options to find out if SpeechRecognition is going work in your particular case.

5.1.1 REQUIRED INSTALLATIONS

SpeechRecognition is the library which is compatible with Python 2.6, 2.7 and 3.3+, but it will require some additional installation steps for Python v2.0. For our project we have used Python v3.0+.
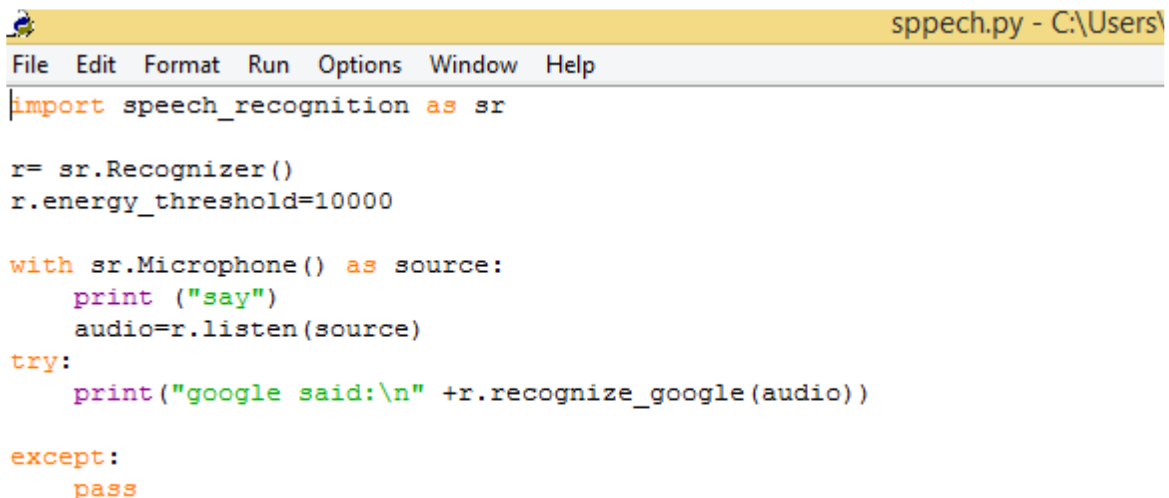
1.>shell-$ pip install SpeechRecognition.

2.>shell-$ pip install python3-pyaudio.

SpeechRecognition will work very good if you need to work with existing audio files. The pyaudio package comes in play when you need to capture microphone input.

The main class which is used in this package is Recognizer class. The use of recognizer instance is obviously to recognize the speech. Every instance of this class comes with various settings and functionality for recognizing speech from the speaker.

```
sppech.py - C:\Users\
File  Edit  Format  Run  Options  Window  Help
import speech_recognition as sr

r= sr.Recognizer()
r.energy_threshold=10000

with sr.Microphone() as source:
    print ("say")
    audio=r.listen(source)
try:
    print("google said:\n" +r.recognize_google(audio))

except:
    pass
```

Figure-7: Speech To Text Demonstration

The Microphone class used in this python program will let the user use the default microphone of their system instead of using some audio files as a source.

If the system of the user doesn't have the default microphone or in case they want to use some other microphone then they will need to specify which one to use by giving a
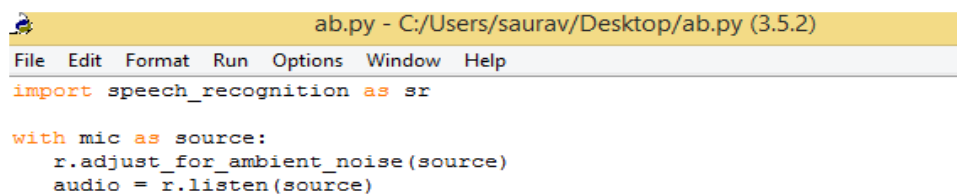
device index. The list can be seen by calling list_microphone_names() which is static method of Microphone class.

Every instance of Recognizer class has seven methods for recognizing speech from speaker source using various APIs:-

• recognize_bing(): Used in "Microsoft Bing Speech"

• recognize_google(): Used in "Google Web Speech API"

• recognize_google_cloud():Used in "Google Cloud Speech" - requires installation of the google-cloud-speech package

• recognize_houndify(): Used in "Houndify by SoundHound"

• recognize_ibm(): Used in "IBM Speech to Text"

• recognize_sphinx():Used in CMU Sphinx - requires installing PocketSphinx

• recognize_wit(): Used in "Wit.ai"

listen():- It is another function used for capturing microphone input. It works just like the AudioFile class while Microhpone is a context manager. Input can be captured from microphone using listen() method of Recognizer class.The first argument taken by this method is an audio source and it will keep on detecting the audio input until the silence is detected by it.

The audio input is generally mixed with ambient noises which can be handled by using the in-built method of recognizer class adjust_for_ambient_noise().

Figure-8: SpeechRecognition Demonstration

You need to wait for a second or two to adjust_for_ambient() to perform its task and then try speaking "Whatever you want" in your microphone and wait for sometime before returning it to recognize the speech again. It only recognizes the speech for one second and it also give you the option to set the duration for wait time.

5.2 TEXT TO SPEECH IN PYTHON

So we have seen above what is a speech to text converter and the theory behind it but the question arises "how to implement this converter in python" so here we go.

5.2.1Pyttsx
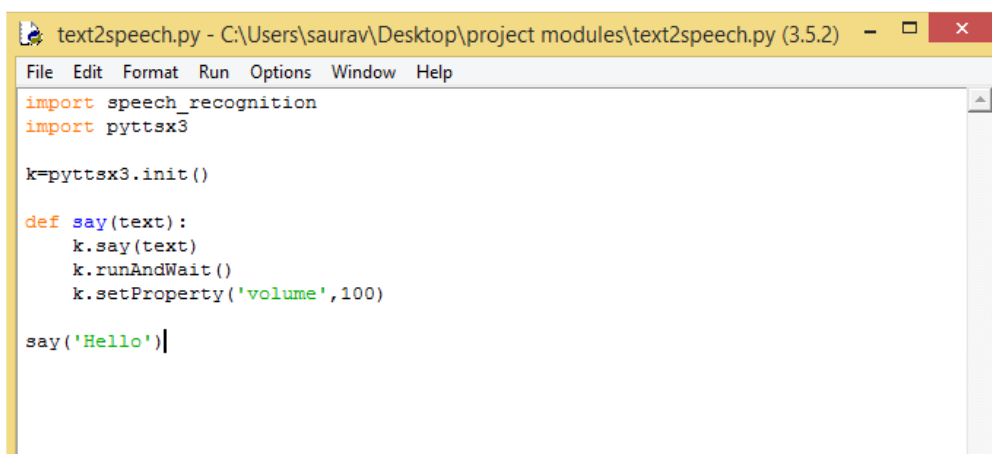
Pyttsx is platform independent that is it is compatible with Windows, Linux, and MacOS speech library. This offers a great set of functionality and features.

The user can set their voice metadata that is information about the data such as gender male or female, pitch of the voice, age, name and language. It supports large set of voices.

So to install it in windows platform depending upon which version of python you are using.

For example if you are using python3 so you need to install pyttsx3.

>>>shell> pip install pyttsx3.


```
text2speech.py - C:\Users\saurav\Desktop\project modules\text2speech.py (3.5.2)
File  Edit  Format  Run  Options  Window  Help
import speech_recognition
import pyttsx3

k=pyttsx3.init()

def say(text):
    k.say(text)
    k.runAndWait()
    k.setProperty('volume',100)

say('Hello')
```
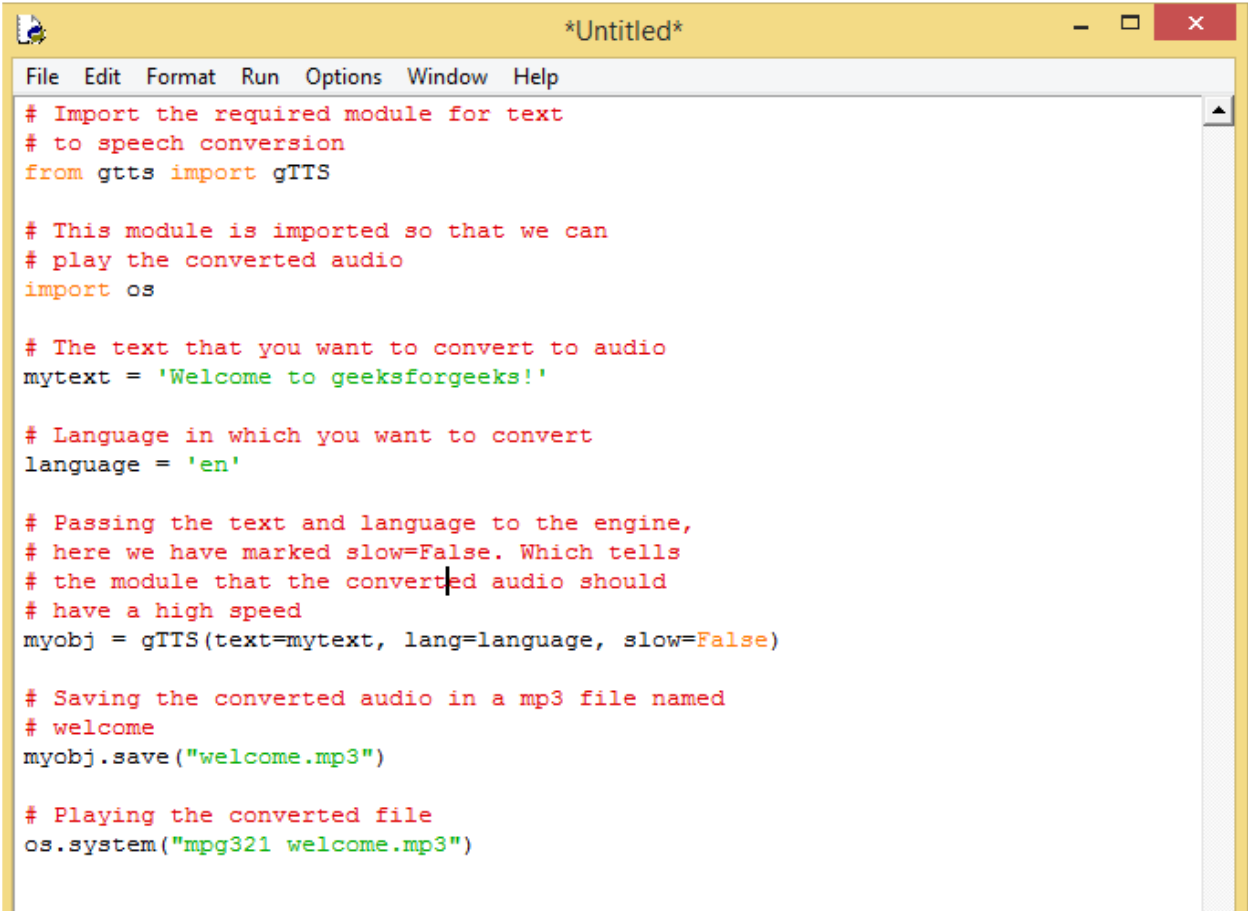
Figure-9: Pyttsx Demo

31

5.2.2 gTTS

Another module which can be used in python for conversion is:-

This module is Google text to speech library in python. gTTS is platform independent that is it is compatible with Windows, Linux, and MacOS speech library. This offers a great set of functionality and features.

To install this API in windows platform

>>>shell>pip install gTTS

```
# Import the required module for text
# to speech conversion
from gtts import gTTS

# This module is imported so that we can
# play the converted audio
import os

# The text that you want to convert to audio
mytext = 'Welcome to geeksforgeeks!'

# Language in which you want to convert
language = 'en'

# Passing the text and language to the engine,
# here we have marked slow=False. Which tells
# the module that the converted audio should
# have a high speed
myobj = gTTS(text=mytext, lang=language, slow=False)

# Saving the converted audio in a mp3 file named
# welcome
myobj.save("welcome.mp3")

# Playing the converted file
os.system("mpg321 welcome.mp3")
```

Figure 10: gTTS Demo

5.3 SIMPLE MAIL TRANSFER PROTOCOL(SMTP)

Email is rising because the one among the foremost valuable service in net nowadays. Most of the web systems use SMTP as a technique to transmit mail from one client to different. SMTP may be a thrust set of rules and is employed to send the mail whereas POP (post workplace protocol) or IMAP (internet message access protocol) square measure accustomed retrieve those mails at the receiver's aspect.

SMTP is Associated with the application layer protocol of OSI model of network.

The user who desires to launch the mail open a TCP (Transmission Control Protocol) connection to the SMTP server and then sends the mail to the other connection. The SMTP server is mostly on listening mode. No sooner the server listens for a TCP connection from any user, the SMTP procedure initiate a connection usually on port number 25. When the successful establishment of TCP connection has been done, the client can send the mail.

The two processes that is sender process and the receiver process carry out a simple request response dialogue, outlined by the SMTP protocol within which the client process transmits the mail address of the mastermind and the recipient for a message. Once the server method accept these mail addresses, the consumer method broadcast the e-mail instant message. The message should include a message header and message text ("body") formatted in accord with RFC 822.

The following example illustrates a message in the RFC 822 message format:

From: yashuchauhan@example.com

To: sauravmishra@example.com

Subject: An RFC 822 formatted message

This is a simple text body of the message.

The blank line separates the header and body of the message.

The SMTP model is of two types :-

1.End-to- end method

2.Store-and- forward method

The SMTP model chains both end-to-end no intermediate message transfer agents and store-and-forward mail delivery methods. The end-to-end method of SMTP is used between organization, and the store-and forward method is chosen for sending mails within organizations which have TCP/IP and SMTP-based networks.

End-To-End

In this method , a SMTP client will speak to the destination host's SMTP server directly to transport the mail. It will keep the mail item from being transmitted until it has been successfully copied to the recipient's SMTP.

Store-and-Forward

In this method a mail can be sent through a number of intermediary hosts, before reaching to the final destination.

A successful transmission from a hosts signify only that the mails has been sent to the next host, and then the mail will be sent to next host.

5.3.1 SENDING EMAIL IN PYTHON USING SMTPLIB

Automation of sending mails using Python can be done by using the smtplib module of Python. Smtplib contains the class SMTP which is useful to connect with mail servers and can be used to send mails. It defines a SMTP client session object which is used to send mail to any internet connected machine that depends on SMTP format.

SMTP is normally used to connect to a mail server and transmit the messages.

The mail server host name and port can be passed to the constructor, or you can use connect() explicitly.

Once connected, just call sendmail() with the envelope arguements and body of the message.

The message text should be a completely created RFC 822-compliant message, since smtplib does not alter the contents of headers.

We have to add header and sender mail and receiver mail by ourselves.

S1 = smtplib.SMTP(h , p , l)

Where h=host name, p=port number,l=localhost name

Host – The argument is used to represent the host which provides you SMTP server. We can specify IP address of the host or a domain name like gmail.com or outlook.com. It is not a compulsory argument.

Port - If the host name is provided then we have to give a port number where SMTP server will listen the requests, normally this port number is 25.

Local hostname - If your SMTP server is running on your local machine, then you can give just localhost in this argument.
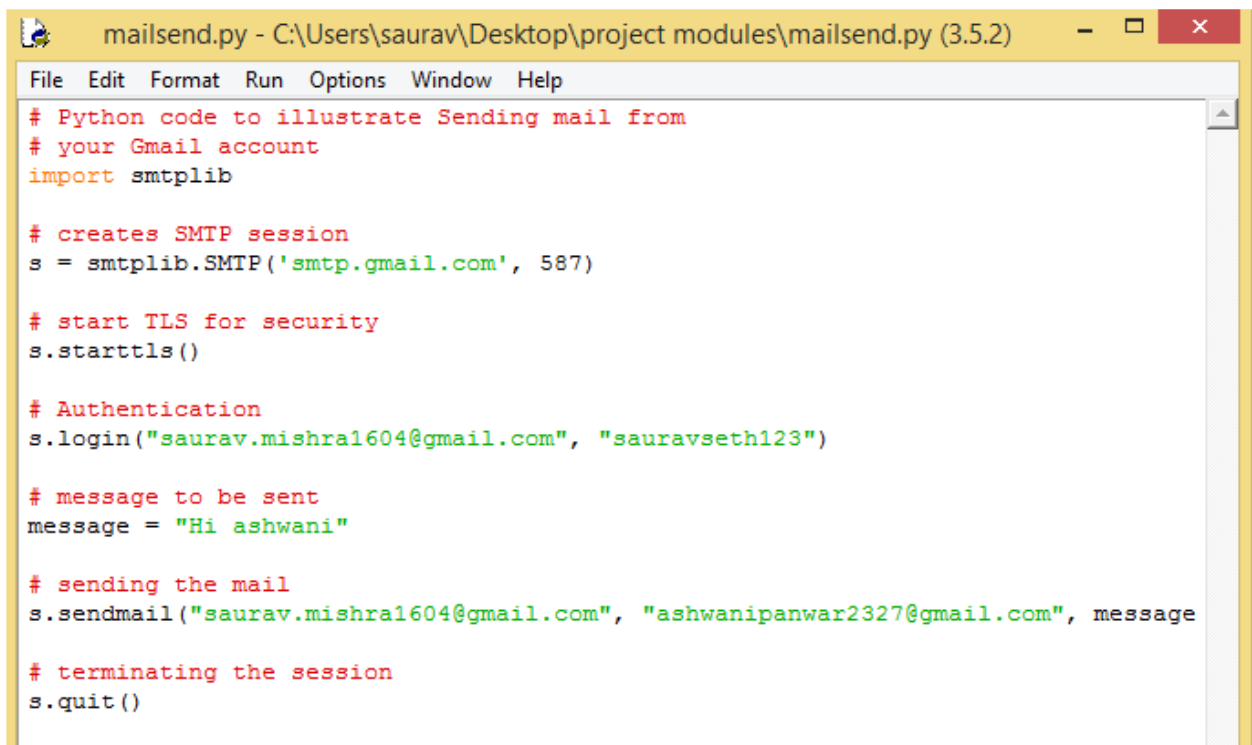
An SMTP object has a method called sendmail, which is usually used to send the mails. It takes following parameters -

The sender - Email-Id of sender.

The receivers - Email-Ids of receivers.

The message - A message arranged like RFC822


```python
# Python code to illustrate Sending mail from
# your Gmail account
import smtplib

# creates SMTP session
s = smtplib.SMTP('smtp.gmail.com', 587)

# start TLS for security
s.starttls()

# Authentication
s.login("saurav.mishra1604@gmail.com", "sauravseth123")

# message to be sent
message = "Hi ashwani"

# sending the mail
s.sendmail("saurav.mishra1604@gmail.com", "ashwanipanwar2327@gmail.com", message

# terminating the session
s.quit()
```

Figure-11: SMTPLIB Code Demonstration

5.3.2 READING EMAIL FROM GMAIL USING PYTHON

This is very important part of our system but python as usual provides very high flexibility in providing this feature. Yes we can automate the process of reading the email from our Gmail account and this can be very useful for the people who can't see so they can use this system to read the email or we can say fetch the unread email from their Gmail account and can listen to it with the help of text to speech converter.

So to achieve our task we just need three modules or functionalities.

1.A mail server and username and its password.

2.Login to Gmail account.(Already discussed how to login using python script) .

3.Servers such as imap.gmail.com and smtp.gmail.com.

4.Most important server needed would be imap.gmail.com.

Imaplib-IMAP4 PROTOCOL

This is the main module which will be used in the process of reading email from your Gmail account using a python script.

Basically this module consists of three classes 1.IMAP4 2.IMAP4_SSL and 3.IMAP4_stream.

These classes are contained in imaplib module whereas IMAP4 is the base class.

All IMAP4rev1 commands are represented by methods of the same name, either upper-case or lower-case. All arguments to commands are converted to strings, except for AUTHENTICATE, and the last argument to APPEND which is passed as an IMAP4 literal. If necessary (the string contains IMAP4 protocol-sensitive characters and isn't enclosed with either parentheses or double quotes) each string is quoted. However, the *password* argument to the LOGIN command is always quoted. If you want to avoid having an argument string quoted (eg: the *flags* argument to STORE) then enclose the string in parentheses)command. Each *data* is either a string, or a tuple. If a tuple, then the first part is the header of the response, and the second part contains the data.

IMAP4.check():-Checkpoint mailbox on server.

IMAP4.close():-Close currently selected mailbox. Deleted messages are removed from writable mailbox. This is the recommended command before LOGOUT.

IMAP4.copy(message_set, new_mailbox):-Copy message_set messages onto end of new_mailbox.

IMAP4.create(mailbox):-Create new mailbox named mailbox.

IMAP4.delete(mailbox):-Delete old mailbox named mailbox.

IMAP4.deleteacl(mailbox, who):-Delete the ACLs (remove any rights) set for who on mailbox.

IMAP4.enable(capability):-Enable capability Most capabilities do not need to be enabled. Currently only the UTF8=ACCEPT capability is supported.

IMAP4.expunge():-Permanently remove deleted items from selected mailbox. Generates an EXPUNGE response for each deleted message. Returned data contains a list of EXPUNGE message numbers in order received.

IMAP4.fetch(message_set, message_parts):-Fetch (parts of) messages. message_parts should be a string of message part names enclosed within parentheses,Returned data are tuples of message part envelope and data.

IMAP4.getacl(mailbox):-Get the ACLs for mailbox. The method is non-standard, but is supported by the Cyrus server.

IMAP4.getannotation(mailbox, entry, attribute):-Retrieve the specified ANNOTATIONs for mailbox. The method is non-standard, but is supported by the Cyrusserver.
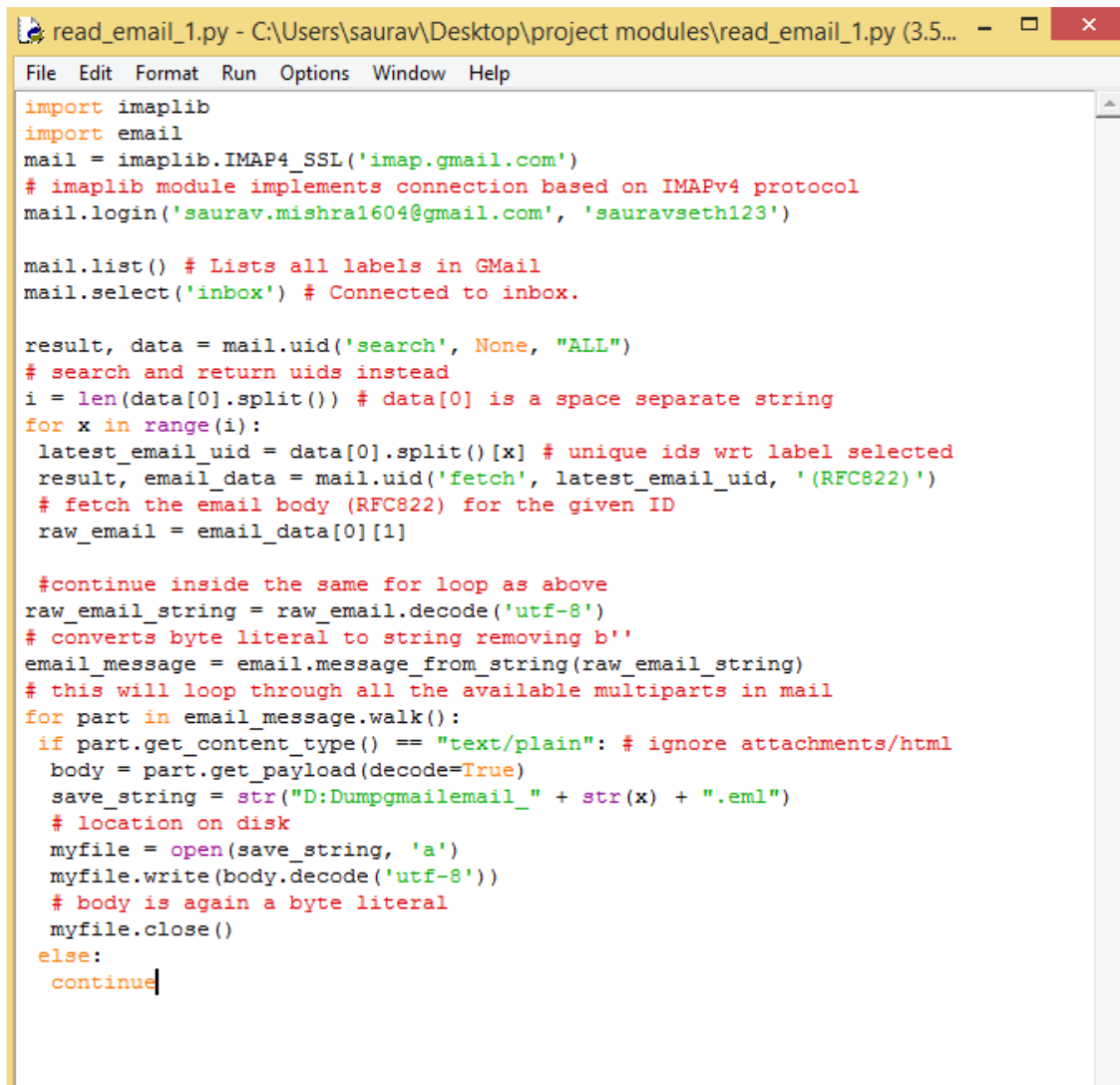
IMAP4.getquota(root):-Get the quota root's resource usage and limits. This method is part of the IMAP4 QUOTA extension defined in rfc2087.

IMAP4.getquotaroot(mailbox):-Get the list of quota roots for the named mailbox. This method is part of the IMAP4 QUOTA extension defined in rfc2087.

IMAP4.login(user, password):-Identify the client using password.The password will be quoted.

IMAP4.login_cram_md5(user, password):-Force use of CRAM-MD5 authentication when identifying the client to protect the password. Will only work if the server CAPABILITY response includes the phrase AUTH=CRAM-MD5.

IMAP4.logout():-Shutdown connection to server. Returns server BYE response.

```python
import imaplib
import email
mail = imaplib.IMAP4_SSL('imap.gmail.com')
# imaplib module implements connection based on IMAPv4 protocol
mail.login('saurav.mishra1604@gmail.com', 'sauravseth123')

mail.list() # Lists all labels in GMail
mail.select('inbox') # Connected to inbox.

result, data = mail.uid('search', None, "ALL")
# search and return uids instead
i = len(data[0].split()) # data[0] is a space separate string
for x in range(i):
 latest_email_uid = data[0].split()[x] # unique ids wrt label selected
 result, email_data = mail.uid('fetch', latest_email_uid, '(RFC822)')
 # fetch the email body (RFC822) for the given ID
 raw_email = email_data[0][1]

 #continue inside the same for loop as above
raw_email_string = raw_email.decode('utf-8')
# converts byte literal to string removing b''
email_message = email.message_from_string(raw_email_string)
# this will loop through all the available multiparts in mail
for part in email_message.walk():
 if part.get_content_type() == "text/plain": # ignore attachments/html
  body = part.get_payload(decode=True)
  save_string = str("D:Dumpgmailemail_" + str(x) + ".eml")
  # location on disk
  myfile = open(save_string, 'a')
  myfile.write(body.decode('utf-8'))
  # body is again a byte literal
  myfile.close()
 else:
  continue
```

Figure-12: IMAPLIB Demonstration

38

5.4 Software Requirements:

Tools Used:

• Python IDLE.

• Interpreters for scripts.

• Selenium Web driver in python.

• Google Speech-to-text and text-to-speech Converters.

• Pyttsx text to speech api in python.

5.5Hardware Requirements:

• Windows Desktop

Chapter 6

TESTING AND RESULTS

6.1 CODE

```
import speech_recognition as sr
import smtplib
import pyaudio
import platform
import sys
from bs4 import BeautifulSoup
import email
import imaplib
from gtts import gTTS
import pyglet
import os, time


print("-" * 60)
print("      Project: Voice based Email for blind")
print("         <--Created by Yashu Chauhan-->")
print("-" * 60)


# project name
tts = gTTS(text="Project: Voice based Email for blind", lang='en')
ttsname = ("path/name.mp3")
tts.save(ttsname)


music = pyglet.media.load(ttsname, streaming=False)
music.play()
```

```python
time.sleep(music.duration)
os.remove(ttsname)


# login from os
login = os.getlogin
print("You are logging from : " + login())


# choices
print("1. composed a mail.")
tts = gTTS(text="option 1. composed a mail.", lang='en')
ttsname = ("path/hello.mp3")
tts.save(ttsname)


music = pyglet.media.load(ttsname, streaming=False)
music.play()


time.sleep(music.duration)
os.remove(ttsname)


print("2. Check your inbox")
tts = gTTS(text="option 2. Check your inbox", lang='en')
ttsname = ("hello.mp3")
tts.save(ttsname)


music = pyglet.media.load(ttsname, streaming=False)
music.play()
```

```python
time.sleep(music.duration)
os.remove(ttsname)
# this is for input choices
tts = gTTS(text="Your choice ", lang='en')
ttsname = ("path/hello.mp3")
tts.save(ttsname)


music = pyglet.media.load(ttsname, streaming=False)
music.play()


time.sleep(music.duration)
os.remove(ttsname)


# voice recognition part
r = sr.Recognizer()
with sr.Microphone() as source:
    print("Your choice:")
    audio = r.listen(source)
    print("ok done!!")


try:
    text = r.recognize_google(audio)
    print("You said : " + text)


except sr.UnknownValueError:
    print("Google Speech Recognition could not understand audio.")


except sr.RequestError as e:
```

```python
    print("Could    not    request    results    from    Google    Speech    Recognition    service;
{0}".format(e))


# choices details
if int(text) == 1:
    r = sr.Recognizer()  # recognize
    with sr.Microphone() as source:
        print("Your message :")
        audio = r.listen(source)
        print("ok done!!")
    try:
        text1 = r.recognize_google(audio)
        print("You said : " + text1)
        msg = text1
    except sr.UnknownValueError:
        print("Google Speech Recognition could not understand audio.")
    except sr.RequestError as e:
        print("Could    not    request    results    from    Google    Speech    Recognition    service;
{0}".format(e))


    mail = smtplib.SMTP('smtp.gmail.com', 587)  # host and port area
    mail.ehlo()  # Hostname to send for this command defaults to the FQDN of the local
host.
    mail.starttls()  # security connection
    mail.login('emailID', 'pswrd')  # login part
    mail.sendmail('emailID', 'victimID', msg)  # send part
    print("Congrates! Your mail has send. ")
    tts = gTTS(text="Congrates! Your mail has send. ", lang='en')
```

```python
    ttsname = ("path/send.mp3")
    tts.save(ttsname)
    music = pyglet.media.load(ttsname, streaming=False)
    music.play()
    time.sleep(music.duration)
    os.remove(ttsname)
    mail.close()


if int(text) == 2:
    mail = imaplib.IMAP4_SSL('imap.gmail.com', 993)  # this is host and port area.... ssl
security
    unm = ('your mail/ victim mail')  # username
    psw = ('pswrd')  # password
    mail.login(unm, psw)  # login
    stat, total = mail.select('Inbox')  # total number of mails in inbox
    print("Number of mails in your inbox :" + str(total))
    tts = gTTS(text="Total mails are :" + str(total), lang='en')  # voice out
    ttsname = ("path/total.mp3")
    tts.save(ttsname)
    music = pyglet.media.load(ttsname, streaming=False)
    music.play()
    time.sleep(music.duration)
    os.remove(ttsname)
    # unseen mails
    unseen = mail.search(None, 'UnSeen')  # unseen count
    print("Number of UnSeen mails :" + str(unseen))
    tts = gTTS(text="Your Unseen mail :" + str(unseen), lang='en')
    ttsname = ("path/unseen.mp3")
```

```python
tts.save(ttsname)
music = pyglet.media.load(ttsname, streaming=False)
music.play()
time.sleep(music.duration)
os.remove(ttsname)
# search mails
result, data = mail.uid('search', None, "ALL")
inbox_item_list = data[0].split()
new = inbox_item_list[-1]
old = inbox_item_list[0]
result2, email_data = mail.uid('fetch', new, '(RFC822)')  # fetch
raw_email = email_data[0][1].decode("utf-8")  # decode
email_message = email.message_from_string(raw_email)
print("From: " + email_message['From'])
print("Subject: " + str(email_message['Subject']))
tts = gTTS(text="From: " + email_message['From'] + " And Your subject: " +
str(email_message['Subject']), lang='en')
ttsname = ("path/mail.mp3")
tts.save(ttsname)
music = pyglet.media.load(ttsname, streaming=False)
music.play()
time.sleep(music.duration)
os.remove(ttsname)
# Body part of mails
stat, total1 = mail.select('Inbox')
stat, data1 = mail.fetch(total1[0], "(UID BODY[TEXT])")
msg = data1[0][1]
soup = BeautifulSoup(msg, "html.parser")
```

```
txt = soup.get_text()
print("Body :" + txt)
tts = gTTS(text="Body: " + txt, lang='en')
ttsname = ("path/body.mp3")
tts.save(ttsname)
music = pyglet.media.load(ttsname, streaming=False)
music.play()
time.sleep(music.duration)
os.remove(ttsname)
mail.close()
mail.logout()
```

6.2 MODIFICATIONS

If you want to save the mp3 files in other directory then just follow the below instruction otherwise don't modify anything:
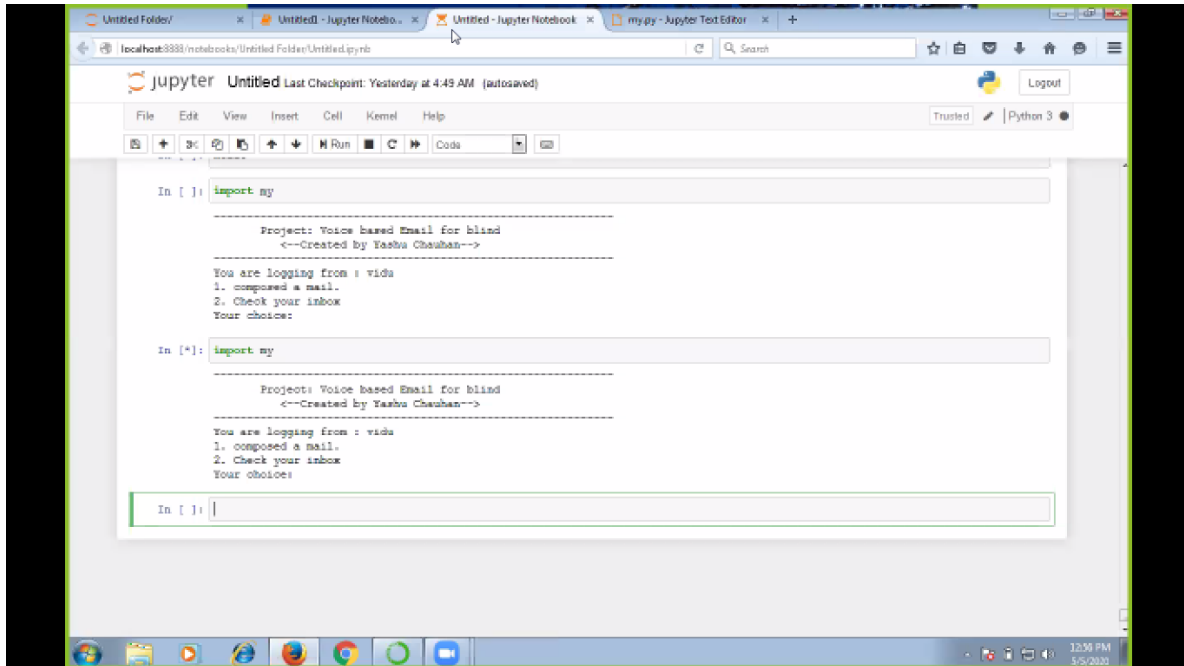
Just add your desktop directory in code wherever the word path has been used in several lines. If you don't know your desktop directory then just open terminal or command prompt and paste the below code. Like: C:\Users\yashu\Desktop (this is my desktop directory).

%userprofile%\Desktop

Also paste your email id, receiver's id, password wherever : emailed,victim and pswrd is written.

If invalid or unsupported audio file occurs or Recall that only FLAC, AIFF, and RIFF WAV files are supported occurs then try this link. Read the file with librosa, then convert it back to a temporary .wav file. Then read it back with the wave package.

6.3 OUTPUT

Figure- 13: Voice Based Email System Demo

Once the user encounters the 'your choice' he will then be allowed to input the action he wants to perform through voice commands and the given command will be executed by the system.

Chapter 7
CONCLUSION

This e-mail system can be used by any user of any age group with ease of access. It has highlight of speech to content just as content to speech with discourse reader which makes planned framework to be taken care of by outwardly hindered individual too. Now the visually impaired people can send and receive mails with a lot of ease only through voice commands without making any use of a keyboard or any mouse. It has helped eradicate the difficulties that the blind people face and made them more the normal individuals.

It has wiped out the idea of utilizing console easy routes alongside screen readers which will help decreasing the intellectual heap of recollecting console alternate ways. Also any non-sophisticated user who does not know the position of keys on the keyboard need not bother as keyboard usage is eliminated. Instructions given by the IVR accordingly to get the respective services offered.

7.1 FUTURE SCOPE

It is a observation that about 70% of total blind population across the world is present in INDIA. This depict the voice message engineering utilized by daze individuals to get to E-mail and multimedia elements of working framework effectively and efficiently. Separated from this the uneducated, crippled and daze individuals will too be able to send sends in their local dialects. This design will likewise decrease intellectual burden taken by blinds to recall and type characters utilizing console. Advances in technology will allow consumers and business to implement speech recognition systems at a relatively low cost and efficiently. Apart from this the system can be enhanced to help the illiterate people by making speech recognition possible in their native languages.

7.2 ADVANTAGES

•The disabilities of visually impaired folks are thrashed.

•This method makes the disabled folks desire a standard user.

•Completely voice based, wiped out the use of keyboard and mouse.

•Efficient and robust

•This design also scales back psychological feature load taken by blind to recollect   and kind characters mistreatment keyboard.

•User friendly

REFERENCES

[1]   Jagtap Nilesh, Pawan Alai, Chavhan Swapnil and Bendre M.R.. "Voice Based System in Desktop and Mobile Devices for Blind People". In International Journal of Emerging Technology and Advanced Engineering (IJETAE), 2014 on Pages 404-407 (Volume 4, issue 2).

[2]   Ummuhanysifa U.,Nizar Banu P K , "Voice Based Search Engine and Web page Reader". In Internationa Journal of Computational Engineering Research (IJCER). Pages 1-5.

[3]   The Radicati website. [Online]. Available: http://www.radicati.com/wp/wp-content/uploads/2014/01/EmailStatistics-Report-2014-2018-Executive-Summary.pdf.

[4]   Geeks for geeks - https://www.geeksforgeeks.org/project-idea-voice-based-email-visually-challenged/

[5]   K. Jayachandran and P. Anbumani "Voice Based Email for Blind People" in International Journal of Advance Research, Ideas and Innovations in Technology(IJARIIT),2017 on Pages 1065-1071

[6]   Pranjal Ingle, Harshada Kanade and Arti Lanke " Voice based e-mail system for blinds" in International Journal of Research Studies in Computer Science and Engineering(IJRSCSE), 2016 on Pages 25-30 (Volume 3, issue 1)

[7]   G. Broll, S. Keck, P. Holleis and A. Butz, "Improving the Accessibility of NFC/RFID-based Mobile Interaction through Learnability and Guidance", International Conference on Human-Computer Interaction with Mobile devices and services, vol. 11, (2009).