

Received : 28th October

Revised : 30th November 2021

Accepted: 15th December 2021

VOICE-BASED EMAIL SYSTEM FOR VISUALLY IMPAIRED

Dr.Y. Adilakshmi^{#1}, Sirazunnisa^{#2}, V. Sai Chand^{#3}, Y. Himaja^{#4}.

#1M. Tech, Ph.D, Associate Professor CSE Department & SR GEC, Gudlavalleru, Krishna

#2Student, CSE Department &SR GEC, Gudlavalleru, Krishna

#3Student, CSE Department &SR GEC, Gudlavalleru, Krishna

#4Student, CSE Department &SR GEC, Gudlavalleru, Krishna

1laxmi072003@gmail.com

2sirazunnisa2000@gmail.com

3saichandvemula99@gmail.com

4himaja.yarrarapu@gmail.com

ABSTRACT:Internet is the most essential part of today's world of communication. E-mails are a further important way of communication that is widely used in the business world. This technology has been useless for incapacitated and blind people. There are around 260 million visually challenged people around the globe and to make this E-mail system closer to visually challenged people, A Voice-Based E-mail System has been proposed. This system provides them the facility of communication by using natural language processing(NLP) packages to convert text-to-speech (TTS) and speech-to-text (STT) so that visually impaired people can operate the system easily. This system reduces the complexity to memorize the characters or information regarding keyboard shortcuts. Every function will be based on simple voice commands so that those people can easily make use of the technologies.

Keywords: E-mails, Natural Language Processing, Speech-to-text, Text-to-Speech, Blind People

I. INTRODUCTION

The internet has become one of the most important tools for everyday life. With the evolution of numerous technologies utilizing the internet, communication has become very simple, and all works can be completed in a shorter time period with more precision and efficiency. Communication is one of the disciplines that has advanced to another level as a consequence of technical breakthroughs and the internet's accessibility. Because of technology improvements, communication has become so simple that distance is no longer an issue.

The first thing that comes to mind when we think of online communication, especially for commercial purposes, is email communication. Email is one of the most reliable and extensively used techniques for sending important information; but, in order to use the internet, one should not be blind. Millions of people in this world who are blind or visually impaired are unable to see the screen or use the keyboards, and hence are unable to use this email system. As a result, the existing email system is worthless for persons who are visually challenged.

Visually impaired and blind persons find keyboard typing challenging because they must rely on others to do so. Because a person must be able to see and read what is printed on the screen in order to access the internet, it is a useless technology for visually impaired persons. There is only one way for a visually impaired person to send an email :They must provide the entire email content to a third party so that the third party can

compose and send it on the visually impaired person's behalf.. However, this method does not lead to a solution to the problem. For maintaining the Integrity of the Specifications, a visually impaired person cannot always find a third person, and the content can sometimes be personal. The optimum solution to this issue isis speech-to-text conversion. Speech recognition can be beneficial to a wide range of people.[4]

As a result, we designed a voice-based email system for the visually impaired, which will assist blind people in sending and reading emails. Users of this system do not need to memorize any basic information about keyboard shortcuts or key locations because our system relies solely on the user's voice commands and does not use keyboards.

Using text-to-speech and speech-to-text conversions, our project aims to make the email system familiar to visually impaired persons. Text-to-speech technology turns text into human-sounding speech and allows you to build your distinctive voice for usage in your apps. Speech-to-text turns the user's voice into text.

II. LITERATURE REVIEW

General Mail System

The existing systems does not provide any voice commands/instructions or audio facilities and therefore it is not suitable for visually challenged people.

Braille keyboard

Impaired people had to remember & recognize the characters of the keyboard which was very difficult on their path. The problem was later solved using Braille keyboard, but these keyboards were very costly.



Screen Readers

Existing screen readers just read the contents on the screen, it doesn't read and send mails particularly and we also need use of keyboard to use screen readers.

To overcome the disadvantages of classical ASR and screen reading systems, a voice-based email system was proposed recently in 2020. The system has advanced capabilities that allow blind people to operate it with convenience. The first module is the Login module, which examines the login credentials. After registering in, the client is taken to the home module, where the following options are available: Inbox, Create, Sent mail, and Junk. In PC program design, IVR technology is used, along with STT (Speech-to-message) and TTS (Text-to-discourse). Mouse click events are also used in the proposed system [1].

The authors proposed a voice-based email system in paper [2], whereby they integrated Google's Gmail. Traditional systems had their email services that have been constructed by the users. The module includes (a) a Speech-to-Text Converter (b) a Speech-to-Text Converter. A text-to-speech system is a device that adherent's text into speech. For sending emails, the application uses the SMTP protocol, and for receiving emails, it uses the POP3 protocol. Speech-to-text accuracy is low since it needs to be educated. It is a desktop application that may be used by people who are illiterate or disabled. The suggested system not only protects the security of the user's data but also gives a feeling of secure mailing to the user.

The researchers of paper [3] proposed an email system that is simple to use for visually impaired people. TTS (Text-to-speech), STT (Speech-to-text), and Mail Software Program (Collaborating, Mailbox, and Then sent Mail) are the three modules that constitute the system design. Speech-to-text is conducted in this system using Artificial Intelligence (AI) via an API involving neural network models given by Google Cloud Speech-to-text to developers. Furthermore, it generates credentials or other credentials into cryptographic algorithms using various Hashing Algorithms (MD5, SHA), resulting in stronger security than older iterations.

III. PROPOSED SYSTEM

The Proposed system is a voice-based email for people with poor eyesight and the user can send and receive emails using only voice commands. It translates the user's spoken speech into text and the text into voice before carrying out the activity. The fundamental benefit of this system is that it does not require the usage of a keyboard; instead, the user will have to answer solely through speech. With good microphone support, we can easily interact with computers and attain our requirements. In this project, each part is designed using different modules in python and some subfields of natural language processing.

The below flowchart shows the execution of this proposed system. when we run this application, a voice prompts the instructions to the user. It prompts them to choose the action which they want to whether it may be read, send, or exited. When the user chooses to send the mail, the voice asks the user to give the details of the mail like subject and body of the mail and audio given by the user is converted into a textual format using speech recognition module and then the mail will be sent and when the user chooses to read the mail, the voice asks the serial number of the mail which they want to read and then the pyttsx3 module converts the text in the mail to the voice format so that the blind person can easily acknowledge the mail. The voice instructions are continuous in a loop until the user asks to exit.

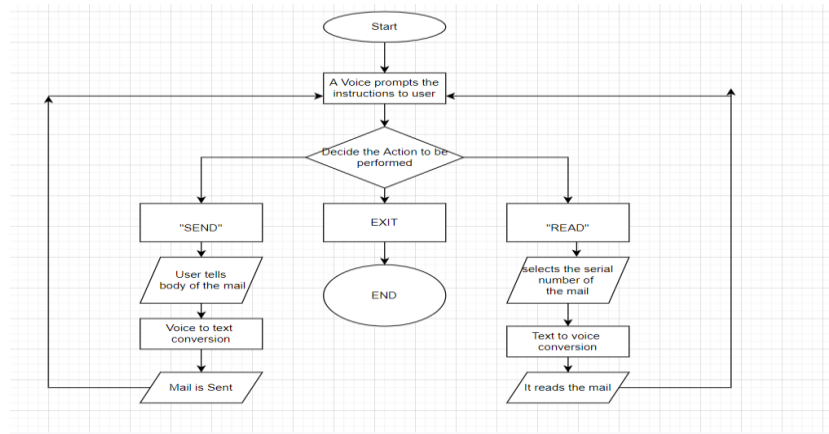


Figure 1: FLOWCHART FOR THIS PROPOSED SYSTEM

The main modules used in the proposed system are:

- Speech Recognition
- Pyttsx3
- Smtplib
- Easyimap

3.1 Speech Recognition

Speech Recognition uses a combination of linguistics and computer science to recognize spoken words and convert them to textual format. The Speech Recognition module enables computers to comprehend human speech. The ability of a machine to listen to spoken words or audio and recognize them is known as speech recognition. The uttered words can then be converted to text, a query can be made, and a response can be given using Python's speech recognition. Some devices can even be programmed to respond to spoken speech. Using computer programs information is taken from the microphone, process it, and convert it into a proper form, you may accomplish voice recognition in Python.

Speech recognition may appear futuristic, but it is already in use. Automated phone calls allow you to shout out your question or the question for which you would like assistance; virtual assistants such as Siri or Alexa also use speech recognition to converse with you.

Python uses linguistic and acoustic modeling methods to accomplish speech recognition. Acoustic modeling is a technique for identifying phonemes/phonetics in a speech to extract the most important parts of speech, such as words and sentences.

With the use of a microphone, speech recognition begins by transforming the sound energy produced by the person speaking into electrical energy. The electrical energy is then converted from analog to digital, and finally to text.

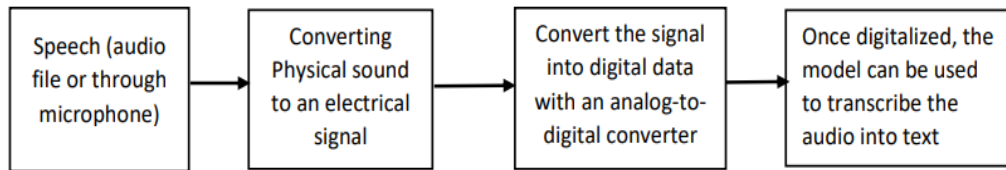


Figure 3.1.1 Working of Speech Recognition

It takes the audio data and breaks it down into sounds, then uses algorithms to analyze the sounds to identify the most likely word that fits the audio. Natural Language Processing and Neural Networks are used to do all of this. Hidden Markov models can be used to enhance accuracy by detecting temporal patterns in speech.

Required Commands to install Speech Recognition module:

- pip install SpeechRecognition
- pipwin install pyaudio

3.2 TEXT TO SPEECH

pyttsx3 is a text-to-speech conversion library written in Python. It works offline, unlike other libraries, and is Python 2 and 3 compatible. An application uses the pyttsx3.init() factory method to get a reference to a pyttsx3. Engine instance. It's a basic program that converts text into speech.

The pyttsx3 module has two voices, one female and the other male, both of which are provided by "sapi5" for Windows.

To get started with installing the pyttsx3 module, open a terminal and type

- pip install pyttsx3

You'll need to install pypiwin32 if you get problems like No module named in32com.client, No module named win32, or No module named win32api.

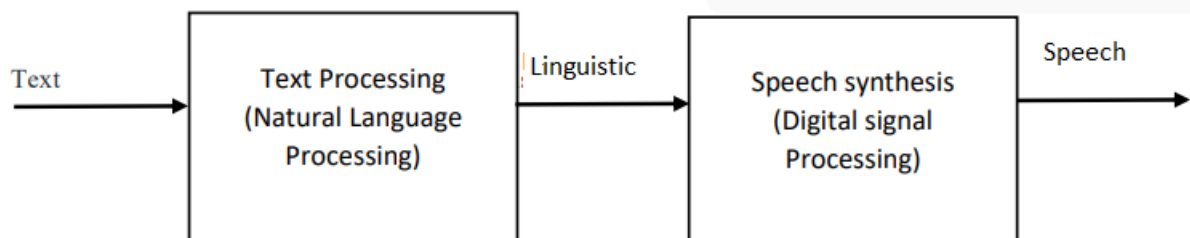


Figure 3.2.1 Text to Speech Conversion using pyttsx3 module

3.3 SMTPLIB

SMTP stands for Simple Mail Transfer Protocol, and it is a protocol for sending and routing emails between mail servers. The `smtplib` module in Python creates an SMTP client session object that may be used to deliver mail to any system on the Internet that has an SMTP or ESMTP listener daemon.

To install smtplib the command is:

- `pip install secure-smtplib`

To send the email, connect to the local SMTP server with the `smtplib` object, and then use the `SendMail` method with the text, from address, and destination address as options (despite the fact that the from and to address are within the e-mail, they are not used to route messages).

If you don't have an SMTP server installed on your local workstation, you can use the `smtplib` client to communicate with a remote SMTP server. Unless you use a webmail service (such as Hotmail or Yahoo! Mail), your e-mail provider should have provided you with the following outgoing mail server information:

- `smtplib.SMTP('mail-domain.com', 25)`

3.4 EASYIMAP

Easyimap is a simple IMAP wrapper. It is used to read and send emails using python. We can also use IMAP library, but it's a long and difficult process, so we'll use easyimap instead, which is simple and quick to install. To install easyimap, enter the command below in your terminal.

- `Pip install easyimap`

To connect to IMAP server, first build an object in which we must specify which server we want to connect to; in this case, we are connecting to the Gmail server.

- `Server_obj = e.connect("imap.gmail.com",username,password)`

To list the ID of the email in our unread inbox, we use **server.listids** and we can create a variable to acquire the email's context, and we can also specify the exact number of the email.

Steps to run the application

- Run the application by double-clicking on it.
- Then a voice command prompt saying to select one action Select required action
 - If action is **“READ”**
 - It asks the serial number of mails which you want to read.

- Then it reads the contents in the mail
- If action is “**SEND**”
 - It asks for the mail id of the receiver
 - It asks to say the body of the mail
 - And then mail will be sent to that receiver.
- If action is “**EXIT**”
 - It Stops the Execution
- If the voice recognizer cannot able to recognize your voice, then prompts you to say again.
- This process is continuous until we give the EXIT command

IV. IMPLEMENTATION

Steps to build the application:

1. Install Python IDLE.
2. Install NLP Libraries.
 - SpeechRecognition and Pyaudio – to convert Speech to text
 - Pyttsx3 – to convert text to Speech
 - Easyimap –to connect to Gmail server and to read/send emails.
3. When we run the application, A voice prompts and guides the user.
4. The Voice prompts operates based on the commands provided by the user.
5. For each command, there is a function associated with it.

4.1. Listen()

- Initially, we should create a recognizer class using **Recognizer()**
- When listen() is called, it takes source using a microphone.
- Using **adjust_for_ambient_noise()** The recognizer class listens to the audio for number of seconds specified from the beginning, then modifies the energy threshold value to make the entire audio more recognized.

- **recognize_google()** method on it to access the Google web speech API and turn spoken language into text. **recognize_google()** requires an argument **audio_data** otherwise it will return an error.

4.2. Speak()

- Initialize **pyttsx3** module using **init()** function.
- **Say()** is used to speak the text.
- We utilize **runAndWait()** to control the speech, Unless the interpreter encounters **runAndWait()**, none of the **say()** texts will be spoken

4.3. readmail()

- This module reads the content in the mail
- To connect to IMAP server, first create an object specifying which server to connect to (in this case, the Gmail Server).

Server_obj = e.connect("imap.gmail.com",unm,pwd)

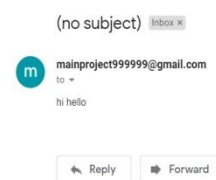
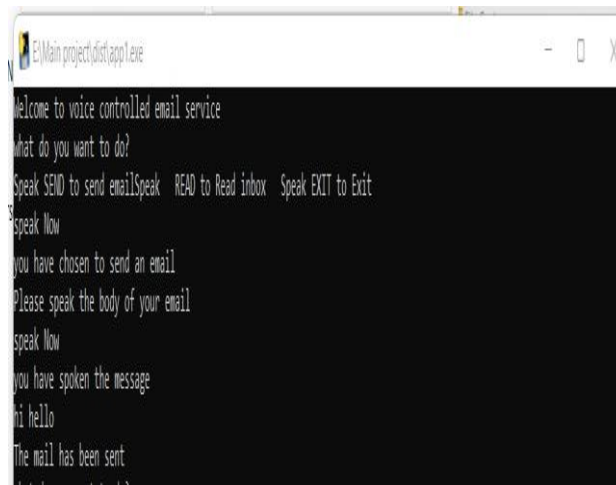
- **Server.listids()**, returns the IDs of the email in our unread inbox.
- **Email.date()** Returns string of 'Date' header.
- **Email.body()** Returns string of Body.
- **Email.title()** returns string of 'Subject' header.

4.4. Sendmail()

- It calls **listen()** function to listen the details of the mail.
- **Server_obj= smtplib.SMTP_SSL("smtp.gmail.com",465)** is used to create SSL connection.
- To login we use **server.login("example@gmail.com", "password")**
- **server.sendmail(unm, rec, msg)** used to send the mail.
- **server.quit()** is used to quit the process of sending of mail

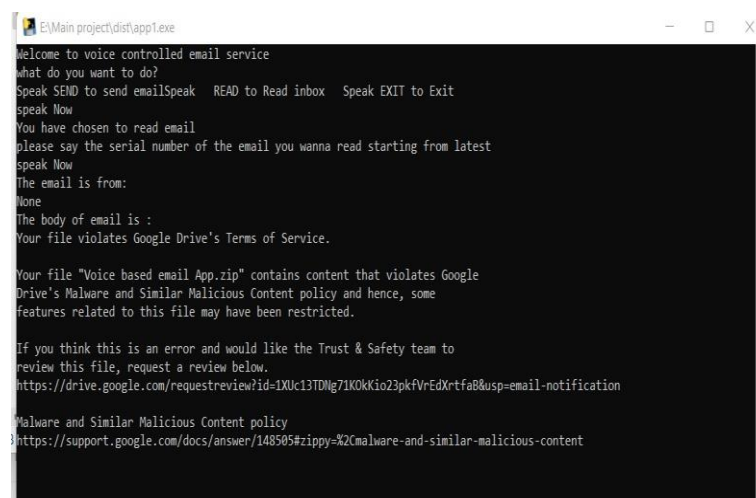
V. RESULTS

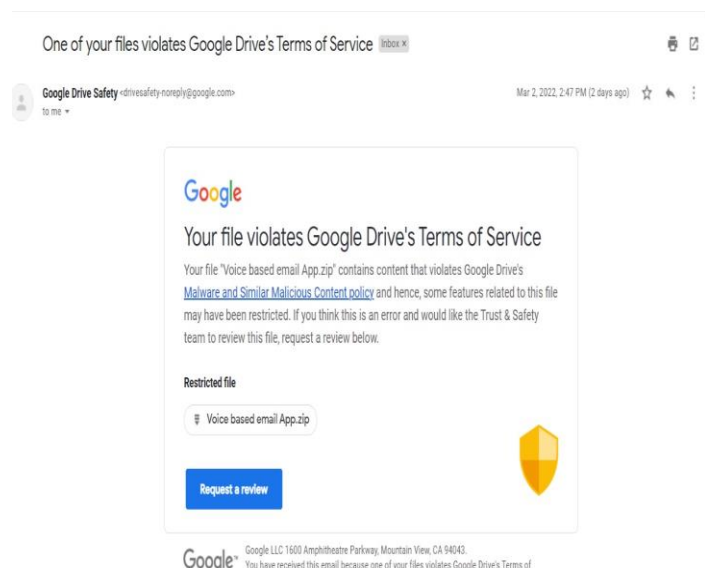
- If we select the “SEND” option



The first image shows that we can send the emails using the proposed system only using voice commands and the second image shows that mail has been successfully sent to the receiver.

➤ If we select the **"READ"** option





In above images, it shows how the read operation has occurred. The voice prompt reads all the content in the mail when a user chooses the read operation.

➤ If we choose the “EXIT” option



When a user chooses the exit command then execution stops. And until you choose exit operation, the voice prompt moves in a continuous loop so that we can read and write any number of mails.

CONCLUSIONS AND FUTURE SCOPE

This project enables persons with visual impairments to participate in the development of digital India and to communicate more easily through the Internet and in people's lives. When you see how to send and receive an email, this technique removes many of the limitations that people have. The developers may be influenced by the project's success, motivating them to produce helpful items that can assist persons with low vision or who are blind.

This system has a lot of potential in the future with a few improvements. This mechanism may also be enhanced to send an attachment, which is particularly useful for people who have a weak vision. It may be made available to anyone in the region and will remain available in a variety of languages

ACKNOWLEDGMENT

I would like to express my deep and sincere gratitude to Dr.Y.Adilakshmi, M.Tech, Ph.D, Associate Professor CSE, Seshadri Rao Gudlavalleru Engineering College, Gudlavalleru for giving me this opportunity to do this project and providing valuable suggestions throughout my project. It was a great privilege and honour to work and study under her guidance.

REFERENCES

- [1] Mamatha, A., Jade, V., Saravana, J., Purshotham, A., & Suhas, A. V. (2020). Voice Based E-mail System for Visually Impaired. *International Journal of Research in Engineering, Science and Management*, 3(8), 51- 54.
- [2] Belekar, A., Sunka, S., Bhawar, N., & Bagade, S. Voice Based E-mail For The Visually Impaired. *International Journal of Computer Applications*, 975, 8887.
- [3] Khan, R., Sharma, P. K., Raj, S., Verma, S. K., & Katiyar, S. Voice Based E-Mail System using Artificial Intelligence.
- [4] Rijwan Khan, Pawan Kumar sharma, Sumit Raj, Sushil Kr. Verma, Sparsh Katiyar "Voice-Based E-Mail System using Artificial Intelligence". In *IJEAT* Volume 09, Issue 03, (February 2020)