Virtual Assistant (Study Bot) in Python

Rucha Uplenchwar[1], Mukta Patil[2], Pratham Gajbhiye[3], Rahul Nagpure[4], Prof. Sangita Jaybhaye[5]

[1,2,3,4] UG Student, Department of Computer Engineering

[5] Faculty, Department of Computer Engineering

BRACT's Vishwakarma Institute of Technology, Pune

(An Autonomous Institute affiliated to Savitribai Phule Pune University)

***Abstract*— Today's technology aims to provide convenience in all aspects of daily life. It can be achieved by automating certain boring or repetitive tasks. Tasks that are performed often can be automated as well, to free the user to do other tasks that can't be automated. This is where a virtual assistant comes into picture. Virtual assistant performs a task or series of tasks with a single command or the click of a button.**

As we know, college occupies a huge portion of a student's life. Online college is slightly hard to manage, with the abundance of lecture links and online material. If some aspects of this college work can be automated, it can take some load off of the students. This is what this virtual assistant aims to achieve. Using advanced python libraries, it performs actions using the user's voice command.

***Keywords*—– Virtual Assistant, Python, Text-to-Speech, Voice Recognition, Artificial Intelligence, Web scraping, Automation**

I. Introduction

Problem Statement

We've all heard of Cortana, Siri, Alexa, Google Assistant virtual assistants that help users with some tasks on the Windows, Android, and iOS platforms. But there is no virtual assistant which is specially developed for students to ease their tasks and make their studies easier.

Purpose

This Software aims at developing a personal assistant for students to make their tasks easy .The software's main purpose is to carry out the user's tasks in response to specific orders, which can be given via speech. It will simplify the user's work by allowing them to finish a task with just one command.

Product Goals and Objectives

Currently, our project seeks to provide Windows users with a Virtual Assistant that will not only assist them with daily tasks such as reporting the time, greeting them and other similar duties, but will also assist them in automating various operations.

So we aim to build a study bot that will make our daily to daily work a lot easier. Using this bot one can do many things through simple voice commands.

II. Literature Review

In daily life there are lots of simple digital tasks that can be automated using voice commands. By doing so, one can focus on the main task and perform multiple tasks at the same time. Voice Assistant/Bot can take the natural voice as input and perform digital tasks.

A. Existing System

Google Assistant

It is developed by Google. It is available on all android devices. The Google Assistant gets activated by saying "Ok Google" or "Hey Google" and you can provide it with a voice input to do the tasks such as Surf the web, open applications, set alarm, send emails and messages, etc.

Apple Siri Assistant

It is developed by Apple Inc. Siri is available only on IOS devices such as iPhone/iPad Siri can perform all the tasks that the google assistant can by taking voice input.

Cortana

Microsoft developed Cortana for Windows Operating System. It helps users to understand and navigate windows in an efficient way. It performs various tasks such as managing the calendar, joining Microsoft teams meetings, setting reminders and alarms, searching the web and opening apps.

B. Proposed System

We are going to build a desktop application assistant using python. The libraries we are going to use are pyttsx3 and speech_recognition. It will take specific voice commands and perform the tasks as per the voice commands. By using this we can open various websites, join lectures, search wikipedia, send mails and get weather reports. With the help of this desktop assistant user can easily and efficiently.

III. System Design

A. Modules Used

i) pyttsx3

It is a text to speech conversion library in python. init() function in this library is used to to get a reference to the pyttsx3.

This package basically contains two voices i.e., male and female voice

- sapi5 for Windows

- espeak for Linux
- Nsss for MAC OS X

To use this library we need to install it first so it can be installed using pip install pyttsx3

ii) speech_recognition

It will return a string output by taking microphone input from the user and that is understood by the machine.
It saves our time by speaking instead of writing
We are going to import speech_recognition as sr so that we may not need to save typing speech_recognition again and again.

Recognizer is a class from this speech_recognition library that converts our speech to text. Recognizer's instance has the seven methods out of which we are going to use the reconginze_google() which works online

To use this library we need to install it first so it can be installed using pip install speech_recognition

iii) datetime

To work with the date as date objects we use the datetime module in python. Date and time manipulation classes are provided by the datetime module.

This module is not just limited to getting the current date and time but for several other operations such as getting a date from a particular timestamp,print current year,day etc.

iv) wikipedia

Wikipedia is one of the biggest and most popular sources for the information on the internet. For fetching the data from the wikipedia we firstly should install the wikipedia library i.e., pip install wikipedia it wraps the official wikipedia API and displays the desired information. This wikipedia module is used to provide easy and simple API to retrieve the information from Wikipedia

v) web_browser

It is a high level interface that allows displaying the web based documents to users. It contains an open() method which accepts the URL as a parameter which in turn opens that link on the web browser. Not only open() but there are other methods such as open_new(),open_new_tab(),etc

vi) os

Interaction between the user and operating system is carried out with the help of os module.

This module is used to carry out various tasks on os whether it is windows,python or linux. Startfile() function is used to open any apps inside the os by giving its path and in result it will open that app for us.

vii) smtplib

smtplib library defines the SMTP client session object that can be used to send mail through the python program. Firstly we are connected to the mail server and then the email is sent through the SMTP protocol. Bunch of methods are available in this module which are used to send emails. To create a mail server SMTP class is used which takes three parameters i.e. host and port.

viii) requests

It is a python module which is used for making HTTP requests to a specified URL. It makes a request to a web page and HTTP returns a response object with all response data

To use this library we need to install it first so it can be installed using pip install requests

ix) Random

A built-in module that we can use to generate and manipulate the random numbers. There are various functions that are present inside the Random module that generates the random number for exa; randrange(x,y),choice([dict]) etc.

x) BeautifulSoup

It is a web scraping framework for python. The data from the web gets retrieved from the website using automated tools which makes the process faster. This python library pulls the data out of XML and HTML files

To use this library we need to install it first so it can be installed using pip install BeautifulSoup

xi) Tkinter

One of the most commonly used methods of creating the Graphical User Interface in python is via Tkinter. It is used for designing the Graphical User Interface for desktop applications. Tkinter is available on various platforms such as Unix,macOS, and on windows systems as well.

xii) PIL(Python Imaging Library)

The Python Imaging Library (PIL) gives image editing capabilities to the Python interpreter.PIL provides powerful image processing capabilities and can handle a wide range of image types such as JPG,JPEG,JPEG 2000,PNG,PCX,MSP etc. PIL and pillow are two different things. Pillow is the fork of PIL library.

.

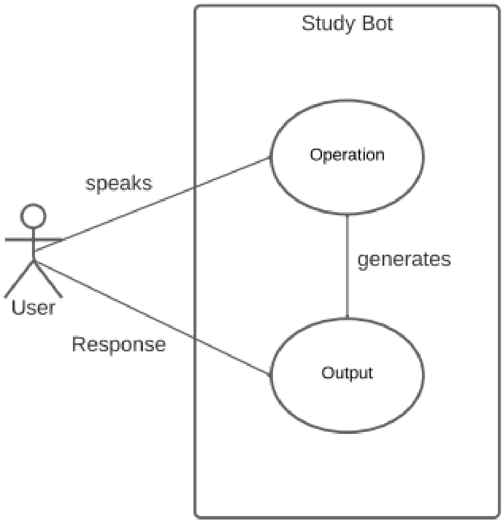
B. Design

The Overall Design consists of the following phases:

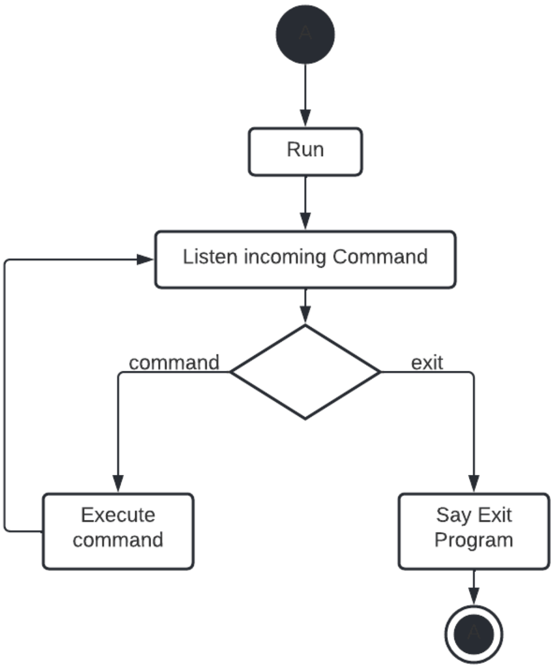
1. Collecting the data in the speech format
2. Interpret the voice and convert it into text
3. Storing and processing the data
4. Speech is generated from the processed text output.

The data will be collected from the user in the speech format and it is stored and used as an input for the next phase. In the next phase the voice will be analysed and further gets converted into the text. The text will be stored and processed which in turn will generate the speech from the processed text output.

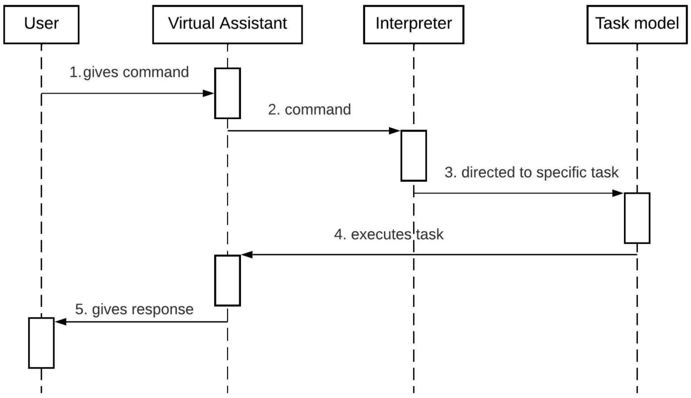
Use case Diagram



Activity Diagram



Sequence Diagram



C. Features

- Open Google
- Open YouTube
- Play songs
- Open Google meet link according subject
- Open Notes folder from Google Drive
- Open Eduplus Portal
- Open VOLP Portal
- Search information on Wikipedia
- Open various apps such as Word, Excel, Powerpoint, Oracle XE
- Send Mails
- Weather Report

IV. Requirements

Software Requirements

- Windows 7/10
- Visual Studio Code
- Python IDE

Hardware Requirements

- RAM – 4 GB or more
- Hard disk – 500 GB or more
- Internet Connection

V. Advantages and Limitations

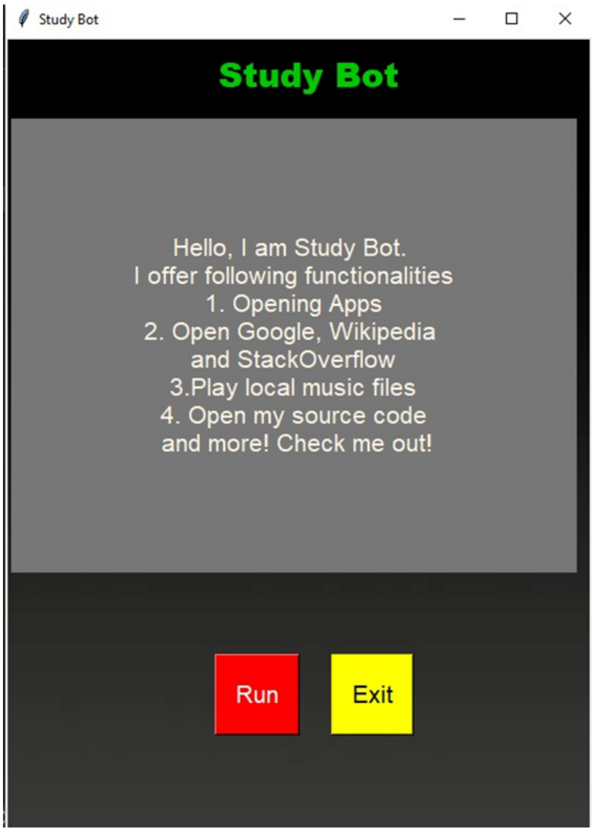
Advantages

1) It automates various tasks
2) It accurately intercepts voice commands
3) It launches various application programs like Word on command
4) It implements basic web scraping to find information from Wikipedia

Limitations

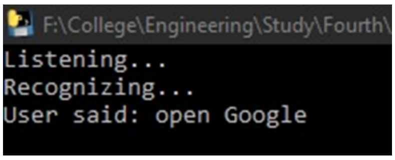
1) It is quite static. Most code is hard coded.
2) It has very little scope for implementation of GUI
3) It implements only basic web scraping
4) It requires internet connection

VI. Results

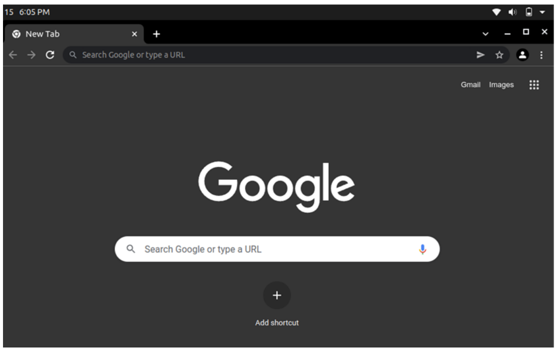


Hello, I am Study Bot.
I offer following functionalities
1. Opening Apps
2. Open Google, Wikipedia and StackOverflow
3.Play local music files
4. Open my source code
and more! Check me out!

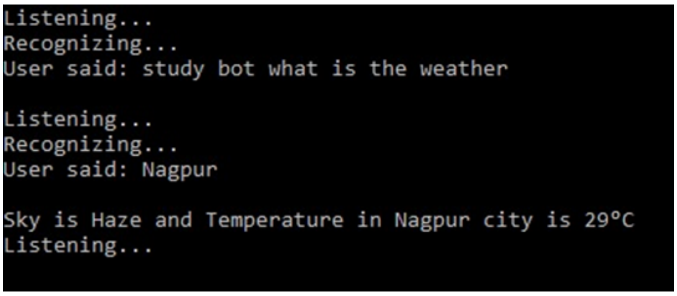
Command: Open Google



```
F:\College\Engineering\Study\Fourth\
Listening...
Recognizing...
User said: open Google
```

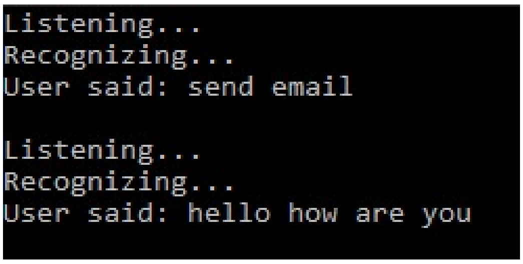


Command: What is the weather?

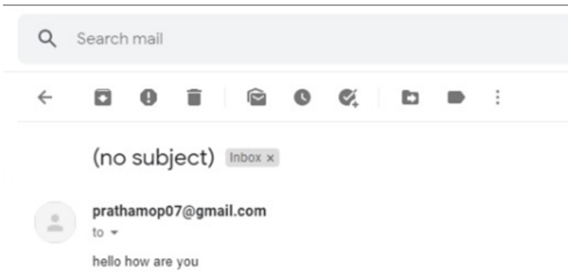


```
Listening...
Recognizing...
User said: study bot what is the weather

Listening...
Recognizing...
User said: Nagpur

Sky is Haze and Temperature in Nagpur city is 29°C
Listening...
```

Command: Send Mail



```
Listening...
Recognizing...
User said: send email

Listening...
Recognizing...
User said: hello how are you
```



(no subject) Inbox ×

prathamop07@gmail.com
to

hello how are you

VII. Future Scope

1) Stopping after every command: instead of actively listening to the user continuously, we can program the application to listen to input after the click of a button. It will make it closely model the existing assistants like Alexa, Cortona and more.

2) Dynamic: Instead of hard coding everything in the application, machine learning can be implemented to make the application dynamic.

3) GUI: User-friendly GUI can be integrated to better show how the application works.

4) Advanced Web scraping: Instead of just showing output from wikipedia, we can code the application to show google search page of given query.

5) Other Languages: We can program the application to take input in languages other than english and still produce the desired result.

6) Pomodoro Timer: We can install a pomodoro timer to run in the background on command for efficient study sessions.

7) Adding more functionalities: More functionalities like opening camera, opening a specific file from system and more can be added.

8) Playing specific videos from youtube: Currently, the system launches the youtube website but cannot play a specific video yet.

VII. Conclusion

We have described a Python-based Voice Personal Assistant (study bot) in this paper. This assistant performs basic activities such as the opening of desktop apps, playing songs, opening google meet links for online lectures, Wikipedia searches, sending emails, weather reports. The current system's capability is limited to just working online. Machine learning will be added into the system in future versions of this assistant, resulting in better suggestions with IoT to control nearby gadgets, similar to what Amazon's Alexa does.

References

[1] Python, Voice Activated Desktop Assistant. n.d. “High Technology LettersISSN NO : 1006-6748.”

[2] Tejaswi, Ch V. 2021. “Virtual Voice Assistant.” International Journal for Research in Applied Science and Engineering Technology. https://doi.org/10.22214/ijraset.2021.35868.

[3] Sermakani., A. M., M. E., J. Monisha, G. Shrisha, and G. Sumisha. n.d. “Creating Desktop Speech Recognization Using Python Programming.” https://doi.org/10.17148/IJARCCE.2021.10325.

[4] *International Journal of Advanced Research in Science, Communication and Technology*. 2022. Naksh Solutions. https://doi.org/10.48175/568.

[5] V., Geetha, C. K. Gomathy, Manasa Sri Vardhan Kottamasu, and Nukala Pavan Kumar. 2021. “The Voice Enabled Personal Assistant for Pc Using Python.” *International Journal of Engineering and Advanced Technology*. https://doi.org/10.35940/ijeat.d2425.0410421.