MINI PROJECT
(2021-22)

Voice Assistant in pc

MID-TERM REPORT

Institute of Engineering & Technology

Submitted by-
Anand Godara
(191500107)
Deevansh Katare
(191500249)
Haresh singh
(191500311)
Rishabh Tripathi
(191500655)
Ritik Varshney
191500662

Supervised By:
Mr Manoj Varshney

Asst. professor
Department of Computer Engineering & Applications

Contents

Abstract

1. Introduction

1.1 General Introduction to the topic

1.2 Area of Computer Science

1.3 Hardware and Software Requirements

Package required

2. Problem statement

Purpose

3. Objectives

4. Implementation Details

5. Some Screenshot

6. References

Abstract

Speech is the most basic means of adult human communication. The basic goal of speech processing is to provide an interaction between a human and a machine.

Speech recognition allows the machine to catch the words, phrases and sentences. Natural language processing is to allow the machine to understand what we speak, and speech synthesis to allow the machine to speak.

In this project we make use of the above feature to create an Automatic Speech Recognition System to make our systems make use of our voice commands to perform real life tasks. For example sending an E-mail, telling what time is it, searching for something on Wikipedia and opening files present on our Systems etc.

Introduction

1.1 General Introduction to the topic

As we know Python is a suitable language for script writers and developers. Let's write a script for Personal Voice Assistant using Python. The query for the assistant can be manipulated as per the user's need.

The implemented assistant can open up the application (if it's installed in the system), search Google, Wikipedia and YouTube about the query, calculate any mathematical question, etc by just giving the **voice command**. We can process the data as per the need or can add the functionality, depends upon how we code things.

We are using **Google speech recognition API** and google text to speech for voice input and output respectively.
Also, for calculating mathematical expression **WolframAlpha API** can be used.
Playsound Package is used to play the saved mp3 sound from the system.

About the Project-

Speech is the most basic means of adult human communication. The basic goal of speech processing is to provide an interaction between a human and a machine.

Speech recognition allows the machine to catch the words, phrases and sentences. Natural language processing is to allow the machine to understand what we speak, and speech synthesis to allow the machine to speak.

In this project we make use of the above feature to create an Automatic Speech Recognition System to make our systems make use of our voice commands to perform real life tasks. For example sending an E-mail, telling what time is it, searching for something on Wikipedia and opening files present on our Systems etc.

How personal assistant works: -

The implemented voice assistant can perform the following task it can open YouTube, Gmail, Google chrome and stack overflow. Predict current time, take a photo, search Wikipedia to abstract required data, predict weather in different cities, get top headline news from Times of India and can answer computational and geographical questions too.

Area of Computer Science

Artificial intelligence (AI) is wide-ranging branch of computer science concerned with building smart machines capable of performing tasks that typically require human intelligence. AI is an interdisciplinary science with multiple approaches, but advancements in machine learning and deep learning are creating a paradigm shift in virtually every sector of the tech industry

Python has a standard library in development, and a few for AI. It has an intuitive syntax, basic control flow, and data structures. It also supports interpretive run-time, without standard compiler languages. This makes Python especially useful for prototyping algorithms for AI.

Hardware Requirements

- Memory [4GB RAM (or higher)]
- Intel core i3 64-bit Processor (or higher)

Software requirements

- Virtual studio code
- Python 3.9.0 64 bits

Package required

- To build a personal voice assistant it’s necessary to install the following packages in your system using the pip command -

- 1) **Speech recognition** — Speech recognition is an important feature used in house automation and in artificial intelligence devices. The main function of this library is it tries to understand whatever the humans speak and converts the speech to text.

- 2) **pyttsx3** — pyttxs3 is a text to speech conversion library in python. This package supports text to speech engines on Mac os x, Windows and on Linux.

- 3) **wikipedia** — Wikipedia is a multilingual online encyclopedia used by many people from academic community ranging from freshmen to students to professors who wants to gain information over a particular topic. This package in python extracts data’s required from Wikipedia.

- 4) **ecapture** — This module is used to capture images from your camera

- 5) **datetime** — This is an inbuilt module in python and it works on date and time

- 6) **os** — This module is a standard library in python and it provides the function to interact with operating system

- 7) **time** — The time module helps us to display time

- 8) **Web browser** — This is an in-built package in python. It extracts data from the web

- 9) **Subprocess** — This is a standard library use to process various system commands like to log off or to restart your PC.

- 10) **Json-** The json module is used for storing and exchanging data.

- 11) **request-** The request module is used to send all types of HTTP request. Its accepts URL as parameters and gives access to the given URL'S.

- 12) **wolfram alpha** — Wolfram Alpha is an API which can compute expert-level answers using Wolfram's algorithms, knowledge base and AI technology. It is made possible by the Wolfram Language.

PROBLEM STATEMENT

We are all well aware about Cortana, Siri, Google Assistant and many other virtual assistants which are designed to aid the tasks of users in Windows, Android and iOS platforms. But to our surprise, there's no such virtual assistant available for the paradise of Developers i.e. Python platform.

PURPOSE

This Software aims at developing a personal assistant for Python-based systems. The main purpose of the software is to perform the tasks of the user at certain commands, provided in either of the ways, speech or text. It will ease most of the work of the user as a complete task can be done on a single command. Jarvis draws its inspiration from Virtual assistants like Cortana for Windows and Siri for iOS. Users can interact with the assistant either through voice commands or keyboard input.

Objective

The main objectives of AI (also called heuristic programming, machine intelligence, or the simulation of cognitive behavior) is to enable computers to perform such intellectual tasks as decision making, problem solving, perception, understanding human communication (in any language, and translate among them).

The present-day pervasiveness of AI, given how little it is noticed in everyday life, suggests that in important ways this objective has been reached.
In the long run, we aim to develop a complete server assistant, by automating the entire server management process - deployment, backups, auto-scaling, logging, monitoring and make it smart enough to act as a replacement

Implementation Details

Part1: - Setting up Search engine

The pyttsx3 module is stored in a variable name engine.

Sapi5 is a Microsoft Text to speech engine used for voice recognition.

The voice Id can be set as either 0 or 1,

0 indicates Male voice

1 indicates Female voice

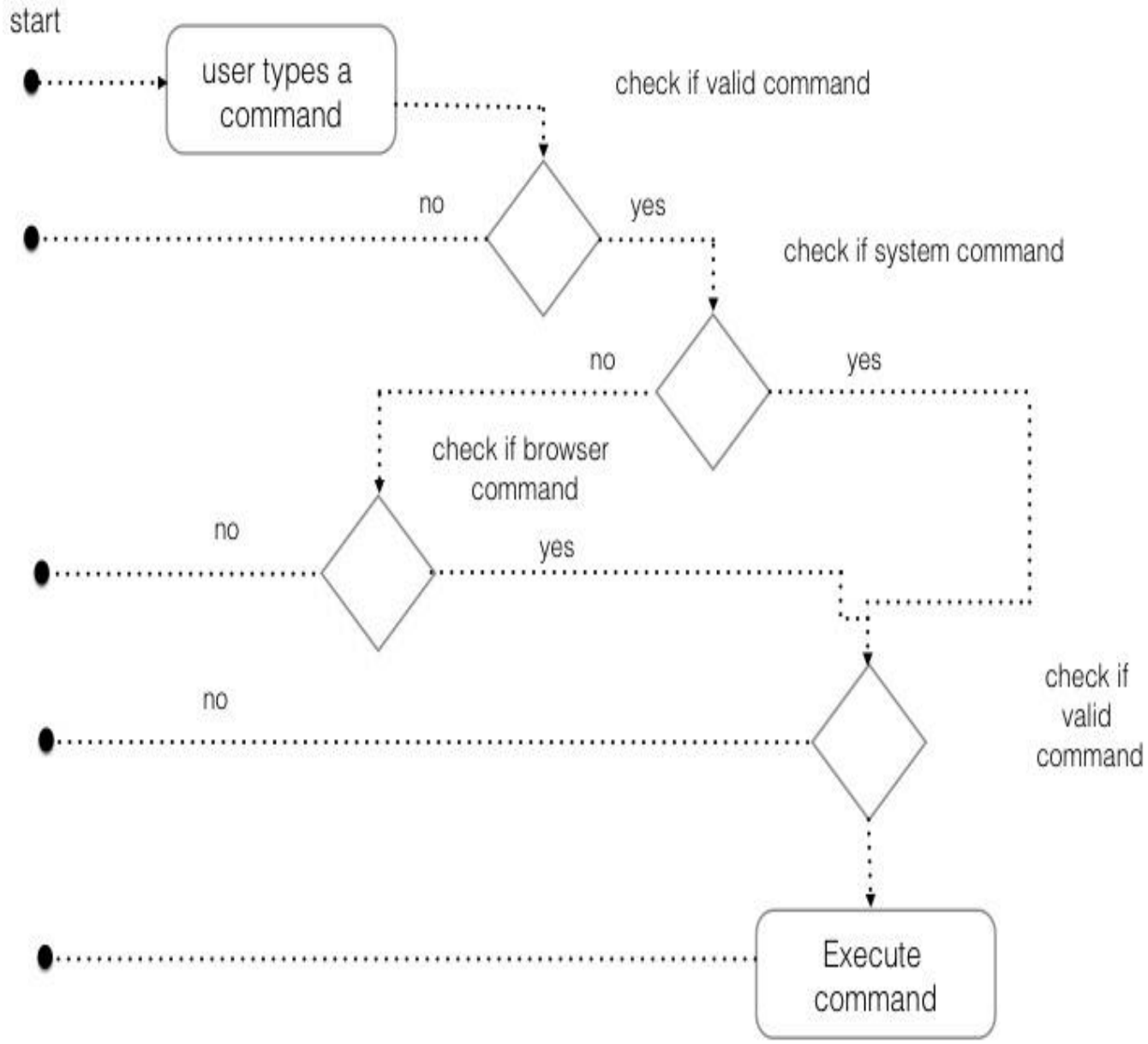
Now define a function **speak** which converts text to speech. The speak function takes the text as its argument,further initialize the engine.

runAndWait: This function Blocks while processing all currently queued commands. It Invokes callbacks for engine notifications appropriately and returns back when all commands queued before this call are emptied from the queue.

Setting up the command function for your AI assistant :

Define a function **takecommand** for the AI assistant to understand and to accept human language. The microphone captures the human speech and the recognizer recognizes the speech to give a response.

The exception handling is used to handle the exception during the run time error
and,the **recognize_google** function uses google audio to recognize speech.


start
user types a command
check if valid command
no
yes
check if system command
no
yes
check if browser command
no
yes
no
check if valid command
Execute command

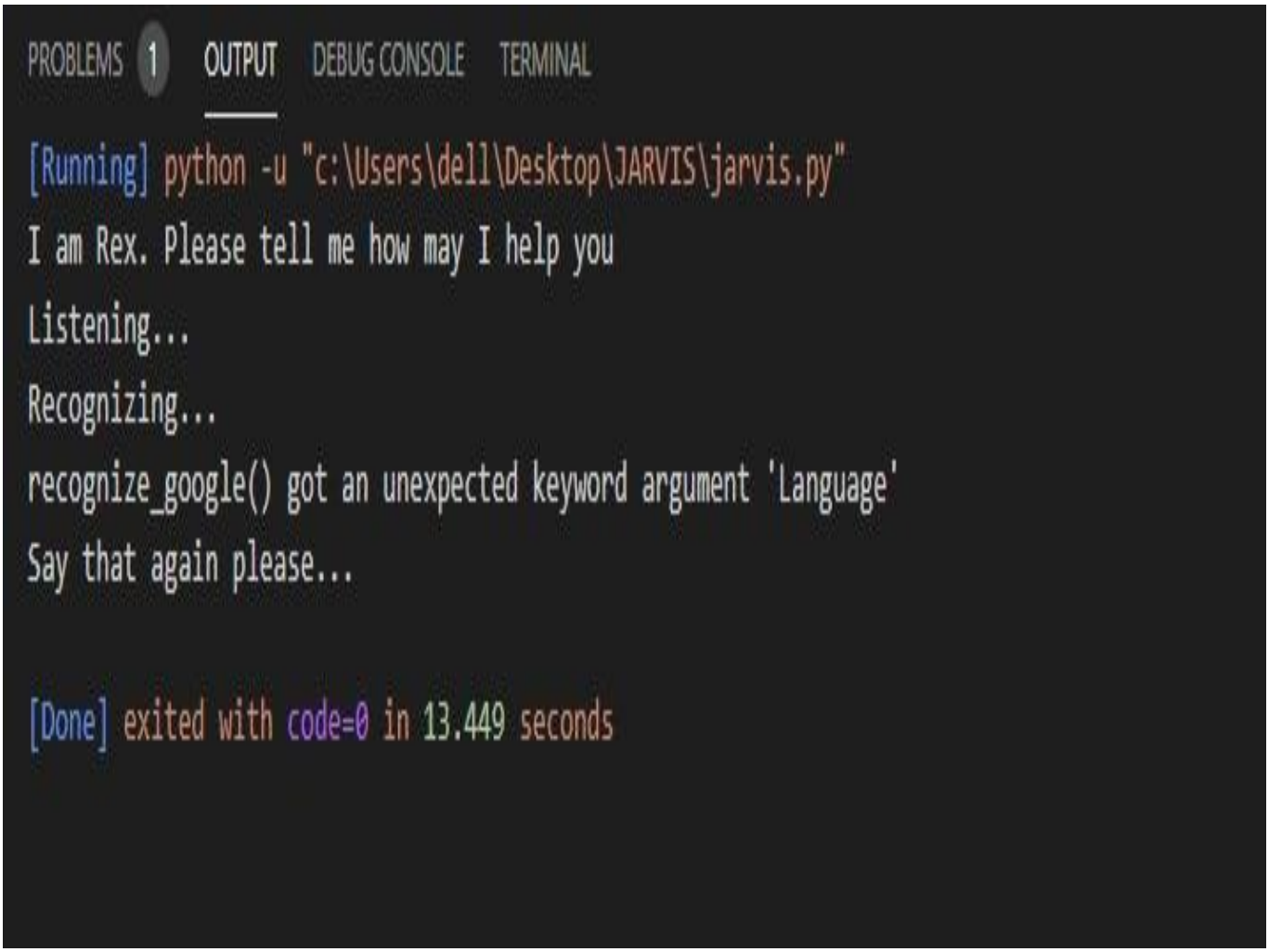
Some Screenshot-

```
import pyttsx3
import speech_recognition as sr
import datetime


engine = pyttsx3.init('sapi5')
voices = engine.getProperty('voices')
#print(voices[0].id)
engine.setProperty('voices',voices[0].id)


def speak(audio):
    engine.say(audio)
    engine.runAndWait()

def wishMe():
    hour = int(datetime.datetime.now().hour)
    if hour>=0 and hour<12:
        speak("Good Morning!")
```

```
    elif hour>=12 and hour<18:
        speak("Good Afternoon!")

    else:
        speak("Good Evening!")
        print("Good Evening!")

    speak("I am Rex. Please tell me how may I help you")
    print("I am Rex. Please tell me how may I help you")

def takeCommand():
    r = sr.Recognizer()
    with sr.Microphone() as source:
        print("Listening...")
        r.pause_threshold = 1
        audio = r.listen(source)

    try:
        print("Recognizing...")
        query = r.recognize_google(audio, Language='en-in')
```

```
    except Exception as e:
        print(e)
        print("Say that again please...")
        return "None"
    return query

if __name__ == "__main__":
    wishMe()
    takeCommand()
```

Output –


PROBLEMS 1
OUTPUT
DEBUG CONSOLE
TERMINAL
[Running] python -u "c:\Users\dell\Desktop\JARVIS\jarvis.py"
I am Rex. Please tell me how may I help you
Listening...
Recognizing...
recognize_google() got an unexpected keyword argument 'Language'
Say that again please...
[Done] exited with code=0 in 13.449 seconds

References-

https://www.youtube.com/redirect?q=https%3A%2F%2Fcodewithharry.com%2Fvideos%2Fpython-tutorials-for-absolute-beginners-

https://www.wikipedia.org/

https://google.com/