AI Desktop Voice Assistance

(Implementation & Upgradation)

Rishav Kumar(2019130)

Prashant Kumar(2019113)

Ank Soni(2019204)

PDPM IIITDM Jabalpur

Overview

First: Introduction

Second: Method Applied/Library Used

Third: Upgradation Being Introduced

Fourth: Results/Conclusion

Abstract

It is never enough to add features in AI-Desktop Voice Assistance. But it has many features like searching in wikipedia, playing music, emailing to someone and many more. The demand for AI-Desktop Voice Assistance is never-ending. So this has many basic features through which we can take it to the next level.

Introduction

The development of artificial intelligence is growing day by day. They are able to recognize human voice as a command, analyse it and provide services to humans. The voice assistant is gaining lots of popularity today. It is the era where no longer a human learns how to communicate with machines, but a machine learns to communicate with a human, exploring his actions, asking about his hobbies, habits and trying to become a human's personal assistant.

A voice assistant is a software program that can perform tasks or provide some kind services for an individual based on the human verbal commands i.e. by using human voice commands and the voice assistant will respond via synthesized voice. The Users can ask their assistants' questions, control home automation devices, and media playback via voice, and manage other basic tasks such as email, to-do lists, open or close any application, send messages to anyone on WhatsApp etc. with verbal commands only.

Methods Applied in the Existing Paper

Solution Approach

The proposed plan started by providing voice input to the voice assistant by the user through a microphone which was later processed and analyzed by voice assistant. The voice input can be anything like getting any information, operation on a computer's internal files etc. Speech recognition has been used to convert which analyses the purpose of the command and calls the required script for execution.

1. Speech Recognition: The system is using Google's online speech recognition system for converting speech input to text.

2. Python Backend: The python backend works on getting the output in exchange of voice input provided by the users through speech recognition module and then identifies whether the command given is Context Extraction, API Call and System call.

3. API Calls: The API will help in communicating with various other softwares. In other words, API is the message passer that delivers the user's request to the provider and then sends the response back to users.

4. Context Extraction: Context Extraction is used to automatically extract structured information from unstructured or semi-structured machine-readable documents.

5. System Calls: It provides an important interface between the process and the OS.

6. Google-Text-To-Speech: Text-To-Speech is basically used for conversion of Speech from Text provided by the user.

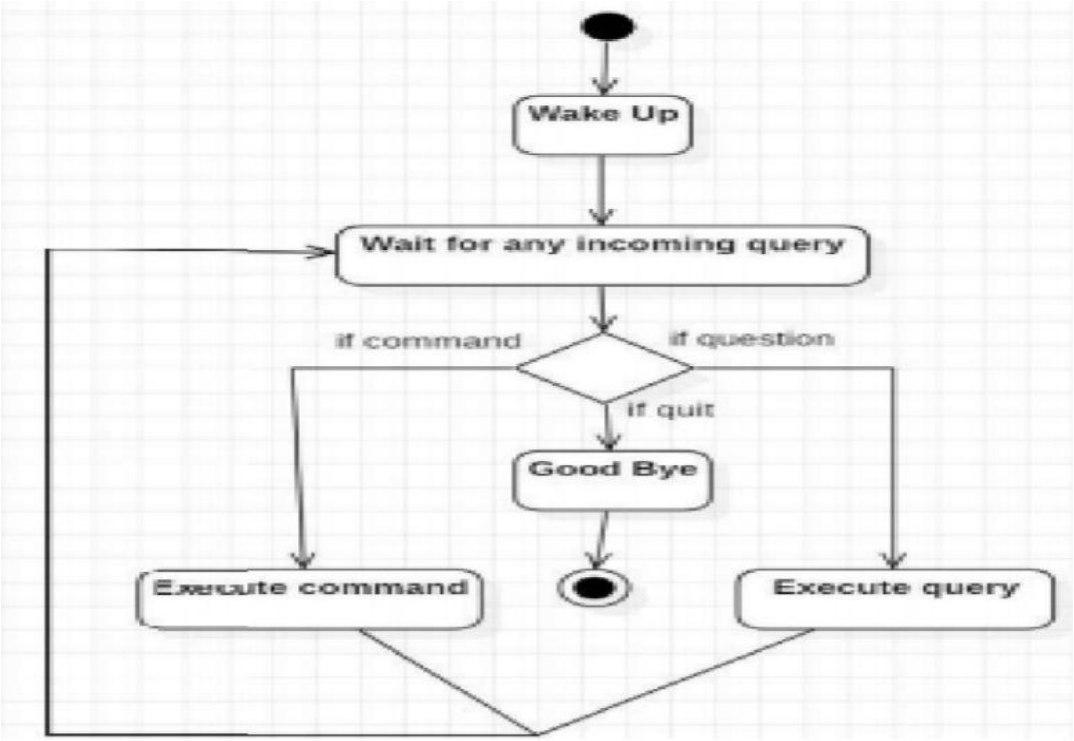
Library Used

- **pyttsx3** :- pyttsx3 is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline and is compatible with both Python 2 and 3. An application invokes the pyttsx3.init() factory

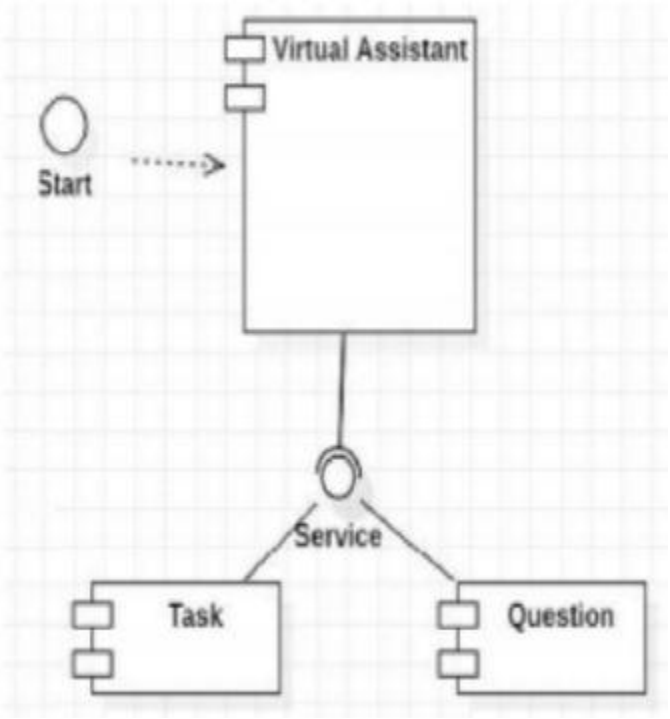
function to get a reference to pyttsx3.Engine instance. It is a very easy to use tool which converts the entered text into speech. The pyttsx3 module supports two voices: the first is female and the second is male which is provided by “sapi5” for windows.

- **speech_recognition**:- speech_recognition is a speech-to-text library, which recognizes the speech/audio using a microphone and returns the string form of data received. This is done with the help of Google Speech Recognition. This requires an active internet connection to work.
- **wikipedia**:- wikipedia is a multilingual online encyclopedia created and maintained as an open collaboration project by a community of volunteer editors using a wiki-based editing system. This library is used to fetch information from wikipedia.
- **webbrowser**:- webbrowser module provides a high-level interface which allows displaying Web-based documents to users. The webbrowser module can be used to launch a browser in a platform-independent manner
- **os**:- The OS module in Python provides functions for interacting with the operating system. OS comes under Python’s standard utility modules. This module provides a portable way of using operating system-dependent functionality.

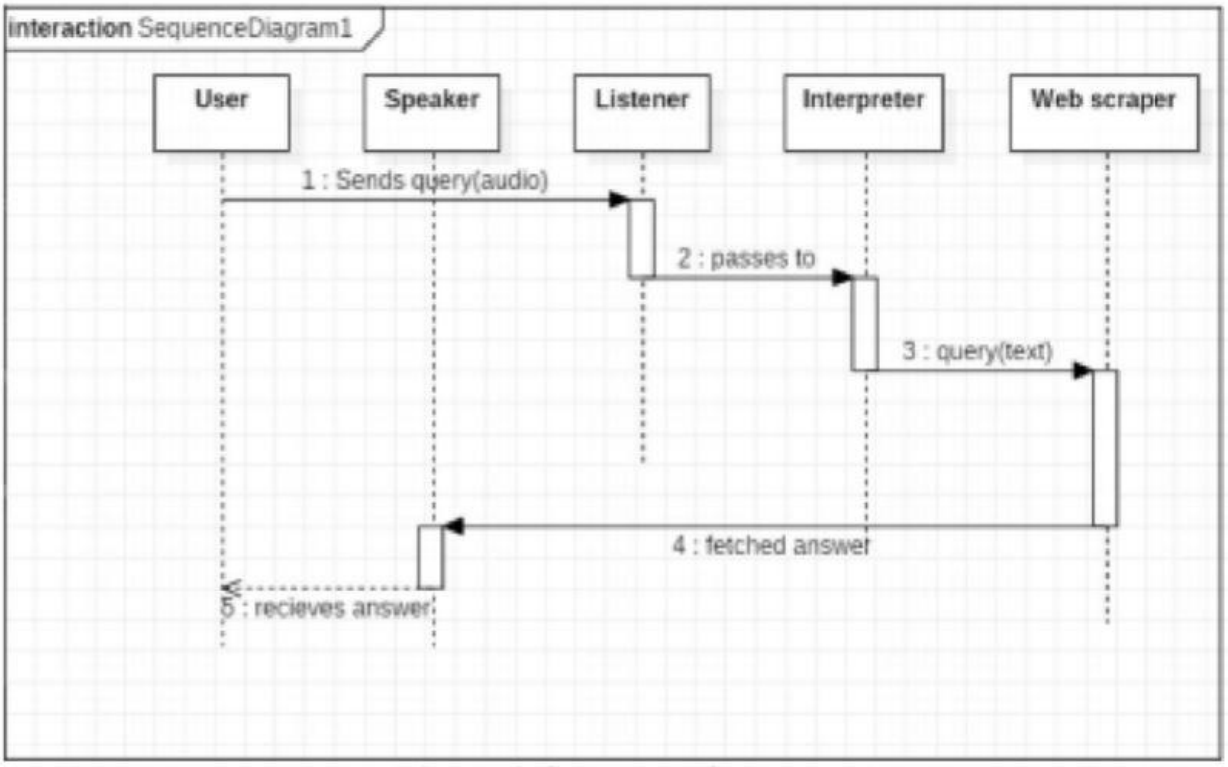
Activity Diagram



Component Diagram



Sequence diagram for Query-Response



Upgradations to be Introduced

Features To Be Added

1. It will ask for your emotion and ask you if it can run the program that we do add to that emotion data.
2. It will also suggest programs that we usually do at that time frame.
3. In the music case it will ask for music taste or play according to your mood.

Results

In this paper we have discussed a Voice Activated Personal Assistant developed using python. This assistant currently works online and performs basic tasks like weather updates, stream music, search Wikipedia, open desktop applications, etc. The functionality of the current system is limited to working online only. The upcoming updates of this assistant will have machine learning incorporated in the system which will result in better suggestions with IoT to control the nearby devices similar to what Amazon's Alexa does.

Conclusion and Future Work

But this is not the end of what AI Desktop Voice Assistants really stands for. The future of these assistants will have the virtual assistants incorporated with Artificial Intelligence which includes Machine Learning, Neural Networks, etc. and IoT. With the incorporation of these technologies, we will be able to achieve new heights. What the virtual assistants can achieve is much beyond what we have achieved till now. Jarvis that we saw in Iron Man is just the motivation to reach there and make it reality. The reach and future of real life Jarvis is still beyond imagination.

References

1. Desktop Voice Assistant Using Natural Language Processing (NLP):- Lalit Kumar
   DOI: https://doi.org/10.46501/IJMTST061262
   Available online at: http://www.ijmtst.com/vol6issue12.html

2. Desktop Assistant AI Using Python:
   Abeed Sayyed1 , AshpakShaikh2 , AshishSancheti3 , Swikar Sangamnere4 , Prof. Jayant H Bhangale5
   Link - http://ijarsct.co.in/Paper1567.pdf

3. Point Matching https://youtu.be/Lp9Ftuq2sVI

4. https://github.com/ggeop/Python-ai-assistant