

# Review on Personal Desktop Virtual Voice Assistant using Python

**Prof. Suresh V. Reddy, Chandresh A. Chhari, Prajwal S.Wakde, Nikhil K. Kamble**

Department of Computer Engineering, SRTTC FOE, MH, India ( Savitribai Phule Pune University)

**Abstract:** In today's develop generation, How cool is it to build your own personal assistants like Alexa or Siri? It's not very complex and may be effortlessly performed in Python. Personal virtual assistants are capturing numerous attentions lately. Chat bots are not unusual in maximum business web sites. The predominant agenda of our voice help makes human beings clever and supply immediate and computed effects. The fundamental mission of a voice assistant is to reduce using enter gadgets like keyboard, mouse, touch pens, and so forth. This will lessen both the hardware fee and space taken by it.

**Keywords** are Voice Assistant, Python, Machine Learning, Text-Speech-Text, Voice Detection and Artificial Intelligence.

## INTRODUCTION

The developments of artificial intelligence (AI) systems which can make virtual human are gaining popularity. Human does not learns to talk with a device, but it's the machine that learns to speak with a humans. Virtual assistants are software program that help you to easily do your day to day works like displaying climate reviews, on off devices and so on. They take input as command through text or through voice. Voice based assistants need a wake phrase to start, followed by the command. There are many voice assistance like Apple's Siri, Amazon's Alexa and Microsoft's Cortana. Such as stream movie on YouTube, Search on Wikipedia and Google, Open websites on web browser, Open applications, Perform windows OS task like Sending and checking mails the voice assistant we've got advanced is a computer-primarily based built using python modules and libraries.

This assistant is just a simple model that might carry out all the simple responsibilities which have been mentioned above however cutting-edge generation is even though right in its miles nevertheless to be merged with Machine Learning and Internet Of Things (IoT) for better improvements. And we also upload GUI to this mission for makes it extra exciting and additionally for person friendly. Personal assistants software program improves person productiveness by means of dealing with routine obligations of the consumer and by using providing records from an internet supply to the consumer. So with a virtual assistant, we are able to be capable of manage many stuff in our desktop/laptop around us with simplest one platform.

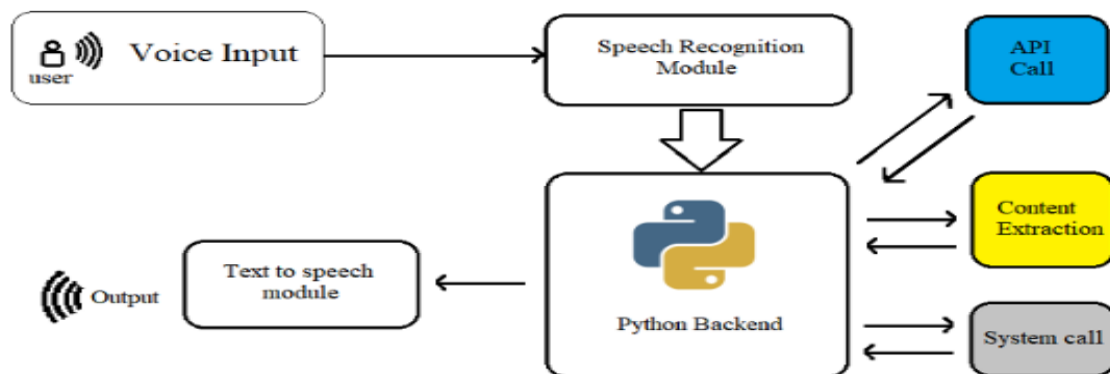


**Figure 1: Timeline of Main Voice Assistants**

## LITERATURE SURVEY

This area of digital assistants having speech reputation has visible some primary advancements or inventions. This is especially due to its call for in gadgets like smart watches or health bands, speakers, Bluetooth earphones, cellular telephones, computer or desktop, TV, and so forth. Almost all the digital gadgets which are coming nowadays with voice assistants, which assist to control the device with speech recognition. A new set of strategies is being evolved constantly to improve the performance of voice computerized seek.

With using voice assistants ,we are able to automate the project without difficulty, simply give the center to the machine within the speech shape and all the duties might be accomplished by means of it from



**Fig 2: Literature view**

content shape to putting off keywords from that text and execute the question to provide outcomes to the person. This has been one of the most beneficial improvements in era. Before AI we have been the ones who have been upgrading technology to do a assignment however now the gadget is itself able to counter new responsibilities and clear up it without need to involve the people to conform it.

## SYSTEM ARCHITECTURE

Imported Python Libraries are:-

```

import speech_recognition as sr
import os
import sys
import re
import webbrowser
import smtplib
import requests
  
```

**Fig 3: Modules Imported**

## SPEECH RECOGNITION:

The voice module used this system is Google’s API i.e. “import speech\_recognition as sr”. This module is used to recognize the sound waves given by the user as input.

This is a loose API this is supplied and supported by Google. This is a totally mild API that facilitates in decreasing the scale of our application.

## TTS & STT

The input voice is first converted to text by using speech recognition module. The text is then processed to result of the

voice by the user. The most time ingesting a number of the STT because the gadget first has to concentrate to the user and unique users have distinctive, a few are smooth to apprehend whilst a few are not without difficulty audible. Once the speech is converted to text executing commands and giving the consequences lower back to the user isn't always a time-eating.

### PYTTSX3:

To convert text into speech in python the pytsx3 module is used. This is an offline module. The module provides run and wait functionality. It is used to allow how much time the system will wait for another input of user. This is a module available in the python community for free that can be installed using the pip command.

### DATETIME:

The Date-Time module is imported to support the date and time. For example, the consumer wants to recognize the modern- day date and time or the person desires to time table a venture at a sure time. In brief this module helps instructions to manipulate date and time and carry out operations according to it handiest. This is a critical module, mainly in tasks in which we need to keep a track of time. This module could be very small in length and allows controlling the dimensions of our program. If the modules are too large or heavy then the system will lag and provide gradual responses.

### WEBBROWSER:

Web-browser module is imported to display information from web to users. If the user wants to open browser and gives input as "Open Google". Then input is processed using this module and the Google browser is opened. The browser which is set in code will open.

### WIKIPEDIA:

Wikipedia is an online library in python which it possible for the virtual assistant to process the queries on Wikipedia and display it to the users. This library needs an internet connection. The number of lines that the user wants to get as a result can be set manually.

### OS MODULE:

OS Module provides operating system dependent functionalities. If we want to perform operations of OS like data reading, data writing, or data manipulate paths then this types of functions are available in an OS module. When the these operations raise an error like "OSError" in case of any error like invalid names, paths, or arguments which may be incorrect or correct but just not accepted by the operating system.

### SMTPLIB:

SMTPLIB is python's standard library which deals with emails. The SMTPLIB library sends mail using "SMTP". This is done using steps that are - initialize, sendmail(), quit. When the optional parameters host and port are provided then connect method is called with these arguments during initialization.

## DESIGN

The design consists of the following:

Taking voice as a input from user.

Conversion of the speech into text by the system

The converted text is then processed to get the desired output.

The text contains some keywords that determine what queries are to be executed. If the keyword did not matched the queries in the program hence the assistant will ask the user to speak again. The output which is in the text form is converted to speech and is provided to the user. Above figure shows the flow of the voice assistant system. speech recognition is used to convert the input voice to text. This text is then sent to the processor, it detects the nature of the command and calls the related code for execution process.

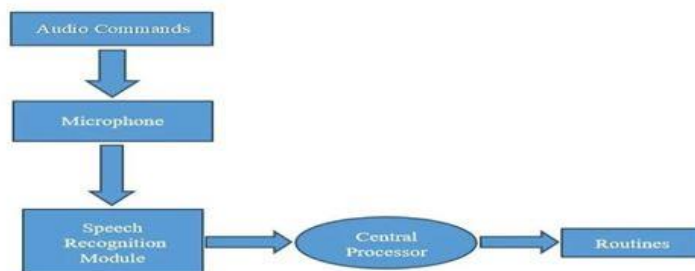


Figure 4:- Block-Diagram How it works

## Literature Review Table:

Sr. no	Paper Title	Year	Methodology	Future scope
1.)	Voice assistant AI-based	2021	“Working of the AI”	We can see the behavioral change .
2.)	Acceptance-Relevant factors of virtual voice assistant	2020	“Listening and recognizing”	All searches across the internet will be voice based.
3.)	Voice assistant and AI’s future.	2019	“Give the user relevant information about his inquiry”	It will increase more personalized experience.
4.)	Desktop speech recognition using python	2018	“Listening and recognizing”	Increase interaction between users and voice assistant.
5.)	AI based voice assistant using python	2017	“It takes the command from the users and work by AI module for give output”	Reduce human efforts and increase personalized experience.

## PROPOSED SYSTEM

This may be whatever like getting movies, opening internal files, and so on. Tests are made via code with the help of books and on line sources, with the aim to find best results and a more expertise of Voice Assistant.

The system has the functionality like:-

The device will hold listening for commands and the time for listening is variable which may be modified consistent with consumer necessities.

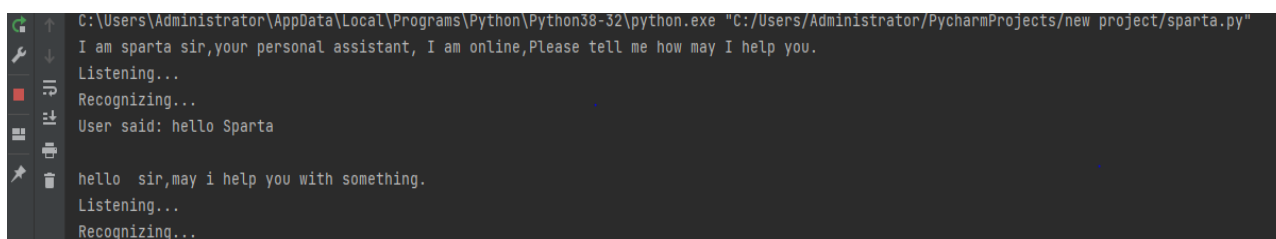
If the system isn't capable of gather facts from the consumer input it will keep asking again to copy until the desired no. of times.

Features supported in the current version include playing song, emails, texts, Wikipedia, OS tasks, or establishing system mounted packages, starting something on the net browser, etc.

The machine will hold listening for commands and the time for listening is variable which can be modified in step with person necessities.

### IMPLEMENTATION AND WORKING MODEL

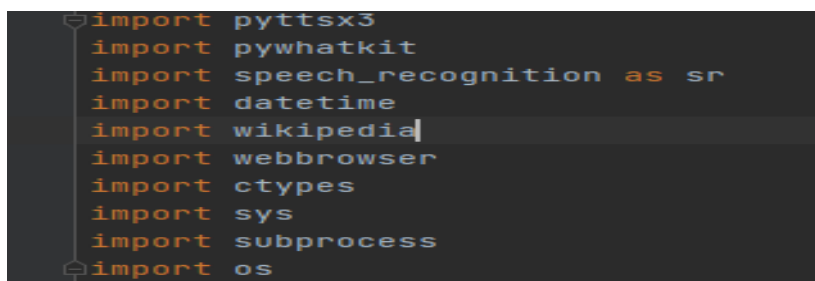
The question for the assistant can be manipulated as per the user's need. Speech attention is the method of changing audio into text. This is normally used in voice assistants like Alexa, Siri, etc. Python presents an API referred to as SpeechRecognition to permit us to convert audio into textual content for similarly processing as shown in given figure. User input:- The assistant will wait for user to give voice command for further process. In figure 5 the assistant introduces itself.



```
C:\Users\Administrator\AppData\Local\Programs\Python\Python38-32\python.exe "C:/Users/Administrator/PycharmProjects/new project/sparta.py"
I am sparta sir, your personal assistant, I am online, Please tell me how may I help you.
Listening...
Recognizing...
User said: hello Sparta
hello sir, may I help you with something.
Listening...
Recognizing...
```

**Figure 5: User Input**

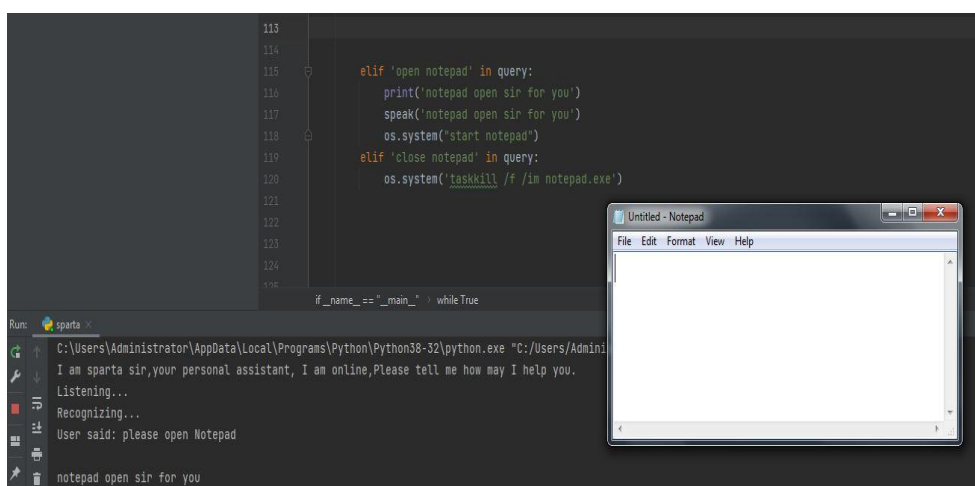
Here we import following some new python libraries. By using these libraries assistant performs functions.



```
import pyttsx3
import pywhatkit
import speech_recognition as sr
import datetime
import wikipedia
import webbrowser
import ctypes
import sys
import subprocess
import os
```

**Figure 6: Python Libraries**

**Open notepad:-** If user gives the command to the assistant to open notepad, then the assistant takes input by listening to the voice of the user and performs operation and gives output and notepad will open for user.

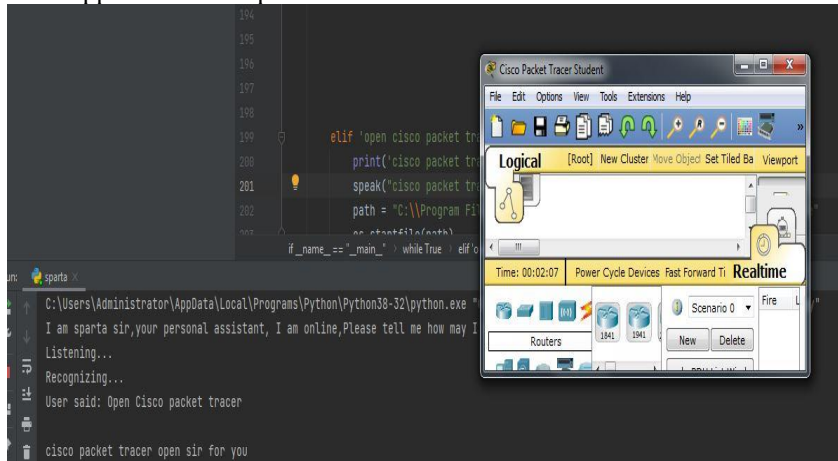


```
113
114
115 elif 'open notepad' in query:
116     print('notepad open sir for you')
117     speak('notepad open sir for you')
118     os.system('start notepad')
119 elif 'close notepad' in query:
120     os.system('taskkill /f /im notepad.exe')
121
122
123
124
125 if __name__ == "__main__": while True
```

```
C:\Users\Administrator\AppData\Local\Programs\Python\Python38-32\python.exe "C:/Users/Admini
I am sparta sir, your personal assistant, I am online, Please tell me how may I help you.
Listening...
Recognizing...
User said: please open Notepad
notepad open sir for you
```

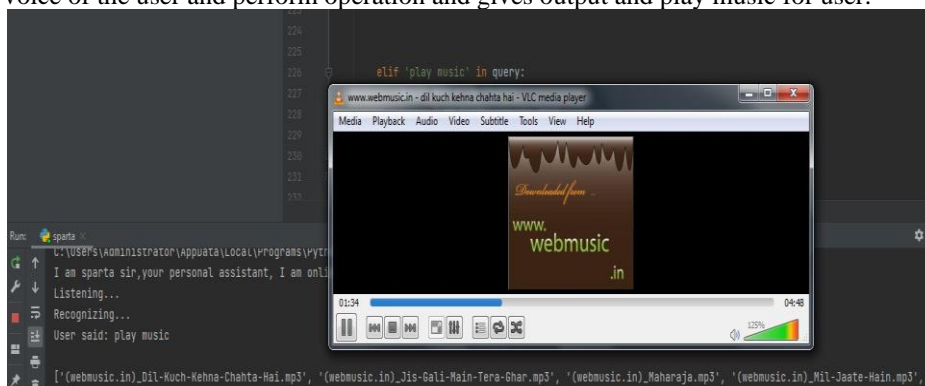
**Figure 7: Assistant open notepad for user.**

**Open application cisco packet tracer:-** If user gives the command to the assistant that open cisco packet tracer, then the assistant takes input by listening the voice of the user and perform operation and gives output and cisco packet tracer application will open for user.



**Figure 8: Assistant open cisco packet tracer for user.**

**Play music:** If user gives the command to the assistant that play music, then the assistant takes input by listening the voice of the user and perform operation and gives output and play music for user.



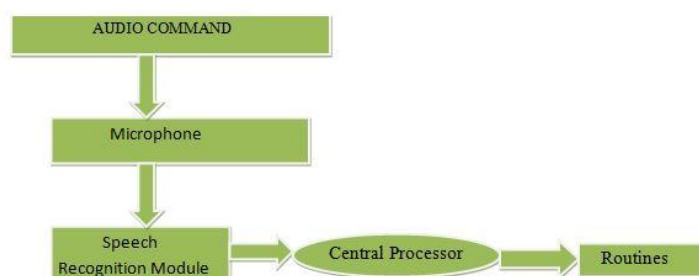
**Figure 9: Assistant play music for user.**

## Algorithm Used

### Natural Language processing (NLP) :-

Natural Language Processing (NLP) is a type of Artificial Intelligence that allows machines to understand and interpret human language. With NLP, machines can understand written or spoken language. It basically convert the text into speech. It also a part of machine learning algorithm.

### Proposed System Architecture



**Figure 10: block diagram that define proposed system architecture**



**CONCLUSION**

In this “Personal Desktop Virtual Voice Assistant using Python” we discussed about the design and implementation of Virtual Voice Assistance. This modules uses open source software py-charm. The nature of this assignment makes it flexible and easy to add additional features without present day machine functionalities. It is not hand work on human commands however additionally it give responses to the user based on the question being requested or the words spoken by way of the user consisting of establishing duties and operations. It is greeting the way the person feels greater comfortable and to interact with the voice assistant. The utility need to also dispose of any kind of useless manual paintings required inside the consumer existence of acting each challenge. The complete machine works at the verbal enter as opposed to the subsequent one.

**FUTURE SCOPE:**

The virtual assistants who might be presently available are speedy and responsive however we nevertheless must go an extended manner. The assistants available these days are still no longer reliable in critical eventualities. The destiny of these assistants may have the virtual assistants integrated with Artificial Intelligence which incorporates Machine Learning, Neural Networks and IoT. With this technology, we are able to gain new heights. What the digital assistants can attain is lots past what we have done till now. Most of us have seen Jarvis, that is a digital assistant evolved through iron guy which is even though fictional but this has set new requirements of what we will achieve using voice-activated digital assistants.

**REFERENCES:**

- R. Belvin, R. Burns, and C. Hein, “Development of the HRL route navigation dialogue system,” in Proceedings of ACL-HLT, 2001
- T.J.Hazen, and L.Hetherington, “JUPITER: A Telephone Based Conversational Interface for Weather Information,” IEEE Transactions on Speech and Audio Processing, vol. 8, no. 1, pp. 85–96, 2000.
- M. Kolss, D. Bernreuther, M. Paulik, Open Domain Speech Recognition & Translation: Lectures and Speeches,” in Proceedings of ICASSP, 2006.
- D. R. S. Caon, T. Simonnet, P. Sendorek, J. Boudy, and G. Chollet, “vAssist: The Virtual Interactive Assistant for Daily Homer-Care,” in Proceedings of pHealth, 2011.
- Crevier, D. (1993). AI: The Tumultuous Search for Artificial Intelligence. New York, NY: Basic Books, ISBN 0-465-02997-3.
- Sadun, E., & Sande, S. (2014). Talking to Siri: Mastering the Language of Apple’s Intelligent Assistant.
- Webopedia, web page  
Available: <https://www.webopedia.com/TERM/I/intelligent-personal-assistant.html>
- Statista, web page Available: <https://www.statista.com/statistics/973815/worldwide-digital-voice-assistant-in-use/>
- Smart sheet web page[Online].  
<https://www.smartsheet.com/voice-assistants-artificial-intelligence>
- Aditya Sinha, Gargi Garg, GouravRajwani, Shimona Tayal, “Intelligent Personal Assistant”. International Journal of Informative & Futuristic Research, Volume. 4, Issue 8, April 2017.
- Emad S. Othman .”Voice Controlled Personal Assistant Using Raspberry Pi”. International Journal of Scientific and Engineering Research Volume 8, Issue 11, November-2017.
- Bibek Behera, “Chappie - A Semi-automatic Intelligent
- <http://en.wikipedia.org/wiki/WebServiceDescriptionLanguage>.
- <http://yudian.voicecloud.cn>
- <http://en.wikipedia.org/wiki/Siri>