

# Python-Based Desktop Assistant with Voice Recognition

Indranil Basu<sup>1</sup>, Saumyadeep Bhattacharyya<sup>2</sup>, Arpan Mondal<sup>3</sup>, Babin Maitra<sup>4</sup>,  
Roshni Joardar<sup>5</sup>, Toyesh Dey<sup>6</sup>, Koushik Pal<sup>7</sup>

Department of Electronics and Communication Engineering<sup>1,2,3,4,5,6,7</sup>  
Guru Nanak Institute of Technology, Kolkata, India

**Abstract:** *Voice assistants offer a more intuitive and efficient means of interacting with computers, reducing the need for input devices like keyboards and mice. By incorporating natural language processing technology, they can understand complex user requests and perform a wide range of tasks. Adding emotion recognition and personalization features can tailor responses to users' preferences and moods. Multi-language support and continuous listening further enhance user experience. Voice assistants have come a long way in terms of development, from the first digital laptop to current devices with impressive processing capabilities. However, there is still potential for further improvement. Voice assistants offer an opportunity to reduce hardware costs and the space required to accommodate input devices. They are a significant breakthrough in human-computer interaction and will continue to become more powerful and efficient as technology evolves. The development of voice assistants reflects the advancements in science and technology. Their applications can extend beyond basic input control and offer users an improved computing experience. Voice assistants will undoubtedly become a key component of our daily lives.*

**Keywords:** Assistant for Desktop, Python Programming Language, Machine Learning Techniques, Text-to-Speech Technology, Speech-to-Text Technology, Language Processing, Voice Recognition System, Artificial Intelligence (AI), Internet of Things (IoT) Integration, Virtual Assistant Technology

## I. INTRODUCTION

In the 21st century, automation is rapidly changing human interaction for greater efficiency. The advancements in technology have led to innovations in Machine Learning, Neural Networks, and other such technologies that allow us to instruct machines to perform tasks or think like humans. One of the most significant innovations in this area is the development of digital assistants, such as Google Now, Siri, and Cortana, which allow users to interact with their computers by simply using their voice.

These digital assistants are particularly useful for elderly individuals, blind and physically challenged individuals, and children who might have difficulty interacting with their computers through traditional input methods. With the help of a voice assistant, even blind individuals can interact with their computers by using their voice alone.

The voice assistant described in this paper is a desktop-based assistant developed using Python modules and libraries. While it is capable of performing basic tasks such as checking weather updates, sending and checking emails, and opening applications, there is still a lot of room for improvement, particularly through the integration of Machine Learning and Internet of Things (IoT).

The future of digital assistants is such that they will be able to execute complex tasks and interact with the user at a more intuitive level, as seen in the fictional digital assistant, Jarvis, in the Iron Man movies. By using voice-activated digital assistants, there will no longer be a need to write long lines of code to perform tasks; the assistant will be able to do it for us. The desktop assistant can operate in supervised, unsupervised, or reinforcement learning modes, depending on its intended use.

With the integration of IoT, the assistant will be able to interact with nearby smart devices and act as a single interface that controls everything in the surrounding environment. Through a successful digital assistant, it will be possible to control multiple devices around us with a single platform.



Fig. 1 Timeline of Main Voice Assistants

## II. LITERATURE SURVEY

The use of digital assistants with speech recognition capabilities has seen significant advancements in recent years, particularly with the rise of smart devices such as smartwatches, health bands, speakers, Bluetooth earphones, laptops, televisions, and mobile phones. As more and more devices integrate voice assistants, there is a growing need for techniques that can enhance the performance of voice-activated search.

To achieve better results, digital assistants need to be trained using Machine Learning and other important techniques such as Artificial Intelligence, Big Data access and management, and the Internet of Things. With the help of these techniques, our devices can learn to adapt to their users' needs and provide personalized results.

Using voice assistants, we can easily automate tasks by providing voice commands, from converting speech to text to extracting keywords and executing queries to provide results to the user. Machine Learning, a subset of Artificial Intelligence, has been one of the most significant advancements in technology, allowing computers to perform new tasks and solve problems without human intervention.

This technology has become an essential part of our daily lives, from personal computers to mobile phones to industrial applications, as these assistants are highly in demand for automating tasks and increasing efficiency.



Fig. 2 Voice Controlled Appliances Affecting Our DailyLife

### III. SYSTEM ARCHITECTURE

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

Fig. 3 Modules Imported

#### 3.1 SPEECH RECOGNITION

Speech recognition is a crucial element of the digital assistant, and for this purpose, we have incorporated Google's Speech Recognition API into our software using the command "import speech recognition as sr" in Python. This API is a widely used and freely available tool that allows the software to recognize and understand the user's voice commands. It is a lightweight tool that significantly reduces the size of our application and enhances its efficiency.

#### 3.2 TTS & STT

Initially, the user's voice input is transformed to textual content through the speech recognition module, which is the first step in the process. The textual content produced by this module is then processed to generate the outcome of the user's query. The last stage in the process is the conversion of the result of the processed query back into speech, which serves as the final output for the user. It is important to note that the most time-consuming of these two stages is the Speech-to-Text (STT) conversion since the computer has to first listen to the user, and different users may have varying levels of clarity in their speech. This is the critical stage that determines the entire execution time. However, once the speech is transformed to text, the task of executing commands and presenting results to the user is not as time-consuming.

Moreover, recent advancements in speech recognition technology have led to significant improvements in accuracy and speed. Neural network-based models such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and their variants are currently the state-of-the-art methods in speech recognition. These models have the ability to learn and adapt to different accents, languages, and background noises, making them ideal for use in digital assistants. Additionally, cloud-based speech recognition APIs, such as Google's Speech Recognition API, provide a convenient and cost-effective solution for developers to incorporate speech recognition into their applications. The use of such APIs can significantly reduce the amount of time and resources required to develop a speech recognition system from scratch.

#### 3.3 IMPORTED MODULES

##### A. PYTTTSX3

The Pytttsx3 is a Python module used for converting text to speech offline, and it is compatible with both Python 2 and 3. It includes run and wait functions, which determine the time interval between inputs. This module is freely available in the Python community and can be installed using the pip command.

##### B. DATETIME

The DateTime module is imported to manage date and time functionality, such as obtaining the current date and time or scheduling tasks at a specific time. This module is essential in tasks that require time tracking and is small in size to optimize program performance.

**C. WEBBROWSER**

The WebBrowser module allows the system to display web-based information to users. For instance, when a user requests to open a website by saying "Open Google," the module processes the input and opens the website on the default browser.

**D. WIKIPEDIA**

The Wikipedia module is a library in Python that facilitates digital assistants to handle inquiries connected to Wikipedia and present the outcomes to the user. This library requires an internet connection to fetch results, and users can set the number of lines to be displayed manually.

**E. OS MODULE**

The OS module provides functionalities that interact with the operating system, such as file operations (reading, writing, or managing paths). Any operation that results in an error, such as invalid names, paths, or arguments, generates an "OS Error."

**F. SMTPLIB**

The SMTPLIB module is included in Python's standard library for working with emails and email servers. It defines an object called the "SMTP client session object," which is used to send emails by the user. The process involves three steps: initialize, sendmail(), and quit. During the first step of initialization, the connect method is called with the optional parameters host and port.

**3.4 DESIGN**

Our system design follows these key phases:

- User input is taken in the form of voice.
- The speech is converted into text for processing by the assistant.
- The processed text is used to generate the required output.
- The text contains one or more keywords that determine the type of query to be executed. If the keyword doesn't match any queries in the code, the assistant requests the user to speak again.
- The final result is generated in the form of speech and delivered back to the user.

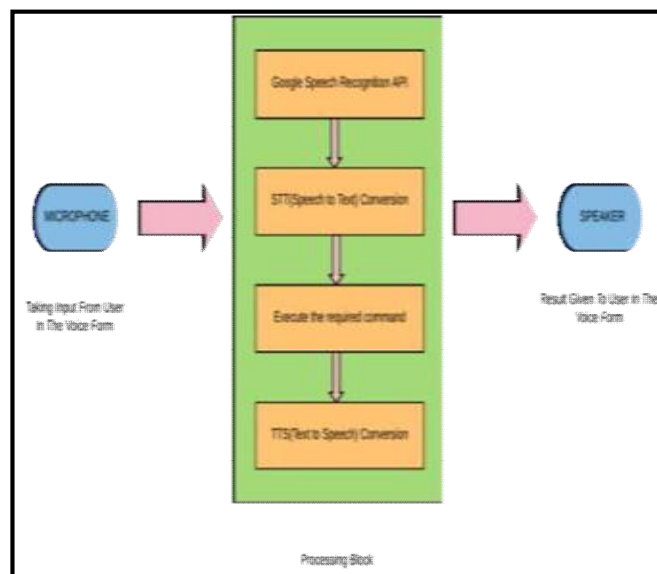


Fig. 4 Processing Block Of STT To TTS

**3.5 PROPOSED SYSTEM**

The proposed system will have the following functionalities:

Copyright to IJARSCT

DOI: 10.48175/IJARSCT-10618

[www.ijarsct.co.in](http://www.ijarsct.co.in)



- The system will continuously listen for user commands with a variable listening time that can be adjusted according to user needs.
- In case the system is unable to receive accurate information from the user, it will keep asking for clarification until the desired number of attempts is reached.
- The system will provide both male and female voices for user selection.
- The system supports features such as playing music, sending emails and texts, searching on Wikipedia, opening device-based applications, opening web browsers, etc.
- The system will continuously listen for user commands with a variable listening time that can be adjusted according to user needs.
- In case the system is unable to receive accurate information from the user, it will keep asking for clarification until the desired number of attempts is reached.
- The system will provide both male and female voices for user selection.

#### IV. FUTURE SCOPE

The current digital assistants are fast and responsive, but there is still a long way to go in terms of their reliability and dependability. Incorporating artificial intelligence technologies such as machine learning, neural networks, and IoT into digital assistants will enhance their reliability and appreciation. With the integration of these technologies, digital assistants can achieve much more than what is possible today. As seen in movies like Iron Man, where Jarvis is a fictional digital assistant, we can set new standards for voice-activated digital assistants.

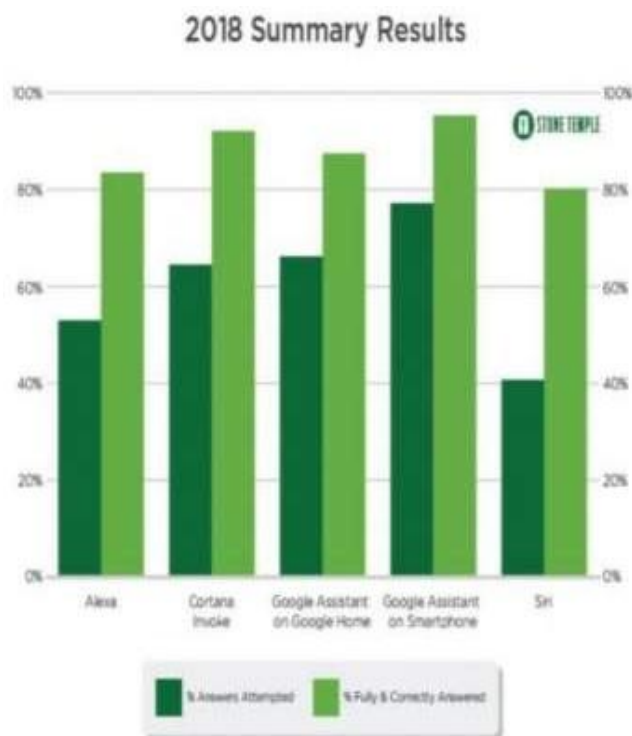


Fig. 5 Accuracy of Results Over Time

#### V. CONCLUSION

In this paper, we have presented a Voice Activated Personal Assistant that was developed using Python. The current version of the assistant performs basic tasks such as providing weather updates, playing music, searching Wikipedia, and opening computer applications. However, the system's performance is currently limited to working online only. Future updates of the assistant will include machine learning algorithms, which will result in better recommendations and the integration of IoT to control nearby devices, similar to Amazon's Alexa

Furthermore, the assistant will be made available offline for tasks that do not require an internet connection.

#### REFERENCES

- [1] D O'SHAUGHNESSY, SENIOR MEMBER, IEEE, "Interacting With Computers by Voice: Automatic Speech Recognition and Synthesis" proceedings of THE IEEE, VOL. 91, NO. 9, SEPTEMBER 2003
- [2] Kei Hashimoto, Junichi Yamagishi, William Byrne, Simon King, Keiichi Tokuda, "An analysis of machine translation and speech synthesis in speech-to-speech translation system" proceedings of 5108978-1-4577-0539-7/11/\$26.00 ©2011 IEEE.
- [3] Nil Goksel-Canbek Mehmet Emin Mutlu, "On the track of Artificial Intelligence: Learning with Intelligent Personal Assistant" International Journal of Human Sciences.
- [4] H. Phatnani, Mr. J. Patra and Ankit Sharma' "CHATBOT ASSISTING: SIRI" Proceedings of BITCON-2015 Innovations For National Development National Conference on Research and Development in Computer Science and Applications, E-ISSN2249-8974.
- [5] Sutar Shekhar, P. Sameer, Kamad Neha, Prof. Devkate Laxman, " An Intelligent Voice Assistant Using Android Platform", March 2015, IJARCSMS, ISSN: 232 7782
- [6] VINAY SAGAR, KUSUMA SM, "Home Automation Using Internet of Things", June-2015, IRJET, e-ISSN: 2395 - 0056.
- [7] "Speech recognition with flat direct models," IEEE Journal of Selected Topics in Signal Processing, 2010.
- [8] Rishabh Shah, Siddhant Lahoti, Prof. Lavanya. K, "An Intelligent Chatbot using Natural Language Processing". International Journal of Engineering Research, Vol. 6, pp. 281-286, 1 May 2017.