STM JOURNALS

Current Trends in
Signal Processing

https://journals.stmjournals.com/ctsp

ISSN: 2277-6176 (Online)
ISSN: 2321-4252 (Print)
Volume 14, Issue 1, 2024
January—April
DOI (Journal): 10.37591/CTSP

Research CTSP

SpeakEasy: Python's Desktop Companion for Effortless Interaction

Abhay Kumar Pandit[1,*], Saurav Kumar[2], Manisha Sharma[3]

Abstract

With their ability to use natural language processing to enable hands-free technological engagement, voice assistants have become an indispensable aspect of modern living. This paper describes how to create a voice assistant with the Python programming language by utilizing its powerful libraries and frameworks for task automation, speech recognition, and natural language processing. The system architecture makes use of several Python modules, including PyAudio, NLTK (Natural Language Toolkit), and speech recognition, to process audio and interpret user inputs effectively. One of the hottest subjects in the modern world is voice assistants, commonly referred to as voice-based artificial intelligence. These are programs that listen to human vocal instructions and reply, allowing for human-computer or device connection. These days, voice assistants are widely available and highly helpful in these hectic times. Due to the global epidemic that has forced people to use smartphones, voice assistants have become ubiquitous today, with Google Assistant being the most widely used. Even 5-year-old children can use it. Alexa from Amazon is a formidable competitor to Google Assistant and can do a wide range of tasks, from providing entertainment to controlling the internet of things devices in homes. One of its best qualities is that it will benefit those with physical disabilities as well. For instance, those who are unable to walk can use the internet of things function to maintain and run household appliances. Thus, we usually try to create a voice assistant that is as user-friendly as the other voice assistants that are popular right now.

Keywords: Voice assistant, speech recognition, text-to-speech, internet of things, Python

INTRODUCTION

Voice assistants, which provide a natural and hands-free method of controlling gadgets and accessing information, have completely changed the way we engage with technology. This paper investigates the creation of a voice assistant using Python, with an emphasis on speech synthesis, linguistic analysis, and speech comprehension libraries. The voice assistant uses Python libraries to offer an approachable and adaptable voice-based interaction solution. For those who are physically disabled, the internet of things (IoT) has become indispensable. Imagine being able to operate your home appliances with simply voice assistance, even if you are unable to walk! It makes daily tasks easy, like having a personal assistant at your disposal. Furthermore, unlike other voice assistants, this one is made to be every bit as stylish and user-friendly as the ones that everyone is talking about [1–4]. The study highlights the voice assistant's potential for a wide range of applications in contemporary computing as it examines the voice assistant's methodology, implementation, and performance.

***Author for Correspondence**
Abhay Kumar Pandit
E-mail: Smartabhay2001@gmail.com

[1,2]Student, Department of Electronics & Communication Engineering, HMR Institute of Technology and Management, New Delhi, India
[3]Associate Professor, Department of Electronics& Communication Engineering, HMR Institute of Technology and Management, New Delhi

Received Date: May 31, 2024
Accepted Date: June 08, 2024
Published Date: June 15, 2024

Citation: Abhay Kumar Pandit, Saurav Kumar, Manisha Sharma. SpeakEasy: Python's Desktop Companion for Effortless Interaction. Current Trends in Signal Processing. 2024; 14(1): 15–23p.

LITERATURE REVIEW

Aarthi Easwara Murthy

Voice-activated personal assistant acceptability of use in the public space was proposed by Easwara

Moorthy et al. (2014); this technology allows users to control home appliances with just their voice, but it may not work as intended when strangers are around. Additionally, voice-activated personal assistants utilize a model that serves as a foundational interpreter and fulfills user requests via the desktop assistant [5].

George Terzopoulos

According to George Terzopoulos et al. (2021), artificial intelligence (AI) is natural language processing, which will be highly beneficial for people in their daily lives. They also suggested voice assistants and smart speakers for use in everyday life and education. People who are visually impaired will find it very helpful as its IoT features enable them to perform daily duties [6].

Rahul Kumar

Rahul Kumar et al. (2020) have proposed a power efficient smart home with a voice assistant. Based on this, we can say that a voice assistant is one of the key components of a smart home, which is quickly gaining popularity in the modern world. Smart locks increase home security, but using a voice assistant to operate home appliances only requires a stable internet connection, which is essential. Additionally, users may accidentally lock themselves out of their own homes [7].

Rajdip Paul

A novel Python-based voice assistance system for reducing the hardware dependency of modern age physical servers was the project that Rajdip Paul et al. (2021) presented. The assistant project that this author has suggested uses Python as the backend and supports system calls, API (application programming interface) calls, and several other functionalities. Although this project responds to api calls very effectively, comprehension and dependability might be used in some work [8].

V. Geetha

A Python-based project called the voice enabled personal assistant for PC was presented by V. Geetha et al. in 2021. With Python serving as the backend, the authors have proposed an assistant project that would allow us to perform tasks like restarting or shutting off our computer with a simple voice command or reading the most recent news. Not every API will be able to translate the raw JavaScript Object Notation (JSON) data into text; however, this project includes a well-supported library. Additionally, the processing of request calls is delayed [1].

Benedict D. C

According to Benedict D. C. et al. (2020), consumer decisions made using artificially intelligent voice assistants are likely to elicit stronger psychological responses from the system when it imitates human behavior. There are IoT functions on the assistant. Although there are certain drawbacks to this paper, it may also order anything that the user desires. To keep up with spoken dialogues, voice assistants rely on the speaker's ability to express the many options. Another significant drawback is that they do not have system calls [9].

Subhash S

A voice assistant powered by artificial intelligence was suggested by Subhash S. et al. in 2020. The speech to text feature has been implemented by the author using a recorded audio file, after which the functions are processed. This project made use of the Google Text to Speech (GTTS) engine and a well-supported library, despite the absence of technologies like IoT and system calls [10].

Nivedita Singh

Following certain API and system calls, Nivedita Singh (2021) et al. presented a voice assistant using the Python speech to text module. This led to the development of a voice assistant using Python that enables the user to issue any kind of command through voice without interacting with a keyboard. Additionally, hybrid platforms can run this. As a result, this article is deficient in certain areas, such as the poorly supported system calls [11].

Dimitrios Buhalis

A paper titled "In-Room Voice-Based AI Digital Assistants Transforming On-Site Hotel Services and Guests' Experiences" was proposed by Dimitrios Buhalis et al. in 2021, where hotel services make use of voice assistants. This will come in very handy in situations like the COVID-19 epidemic. In this COVID era, human contact is seen as dangerous, and losing human touch when using a voice assistant is not seen as advantageous. It can also be used to regulate the lights in the room and the temperature, although complex integration and staff training are required [12].

A.M. Sermakani

The creation of desktop speech recognition using Python programming was suggested by Sermakani et al. in 2021 [2]. With Python serving as the backend, this project combines AI technologies such as voice activation, automatic speech recognition, text-to-speech, and voice biometrics. There is strong brain technology in this project. The flowchart of the proposed system is shown in Figure 1.

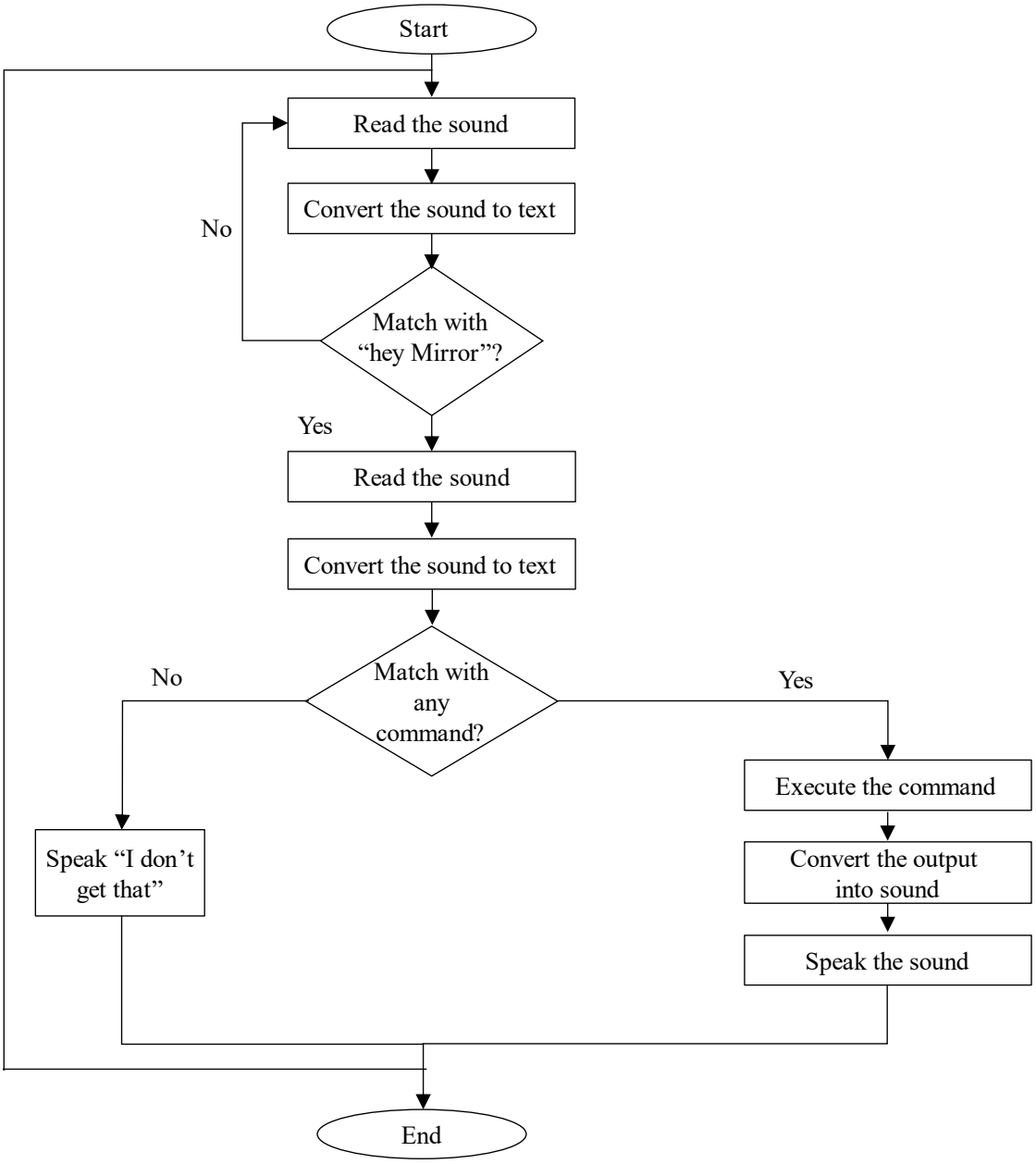

Figure 1. Flowchart of the voice assistant process.

METHODOLOGY

Several crucial processes are involved in the process of creating the Python-based voice assistant, such as choosing the right libraries, creating the architecture, putting the parts together, and testing the functionality.

Selection of Libraries

- Select Python libraries for linguistic analysis, voice synthesis, and speech understanding depending on their features, usability, and suitability for the project.
- For instance, choose the Natural Language Toolkit (NLTK) or spaCy library for natural language processing and the Speech Recognition library for speech recognition.

Designing the Architecture

Create the voice assistant's architecture, considering the parts for conversation control, language understanding, speech synthesis, and speech interpretation. The case diagram of the proposed system is shown in Figure 2.

- Ascertain how these elements will work together to process user input and provide output.

Implementation of Speech Recognition

- Use the chosen library to put voice recognition functionality into practice.
- Set up the library so that it can translate user vocal input into text.
- Use the chosen library to implement speech recognition features.
- Set up the library to interpret user speech input and translate it to text.

Implementation of Natural Language Processing

- Use features for natural language processing to interpret user inquiries.
- Analyze the text input using the chosen natural language processing (NLP) library to extract pertinent data, including entities and user intent.

Implementation of Text-to-Speech Synthesis

- To produce vocal responses, use text-to-speech synthesis functionality.
- To translate written responses into spoken language, use the chosen text-to-speech library.

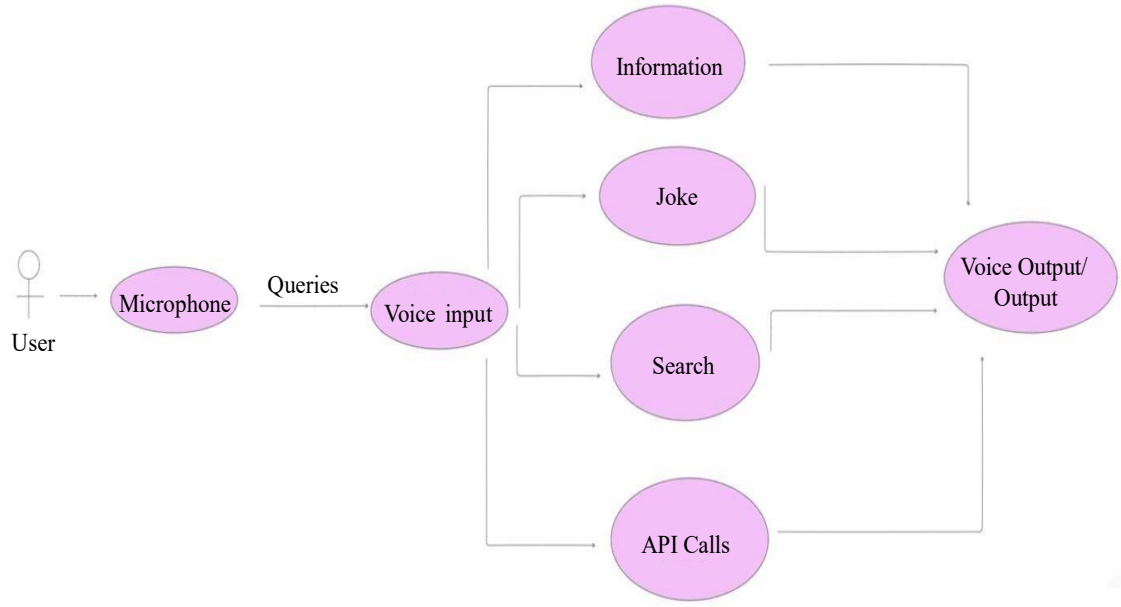

Figure 2. User case diagram of proposed system.

PROPOSED WORK

A voice assistant is one of the most important issue solvers in the current world. Any problem can be quickly solved with just the user's voice; no more information is needed to obtain the answer. It is also helpful for maintaining the homes, like turning on and off a stopwatch. Here, we will attempt to comprehend a use case diagram's design. The following are some explanations of potential system scenarios: As actors in the Use-Case Diagram, each User is represented by a voice assistant, which is a Use Case provided by NOVA. Using the microphone, the user asks a question of the assistant, which the assistant interprets using a speech-to-text module that transcribes the actor's speech into text, processes it further into modules, and then makes the appropriate calls.

Mandatory Features of a Voice Assistant

1. *Have to give the user any information they request:* The user may require any information that is readily available online, but finding and reading such material takes time. Fortunately, we can accomplish the process of obtaining the information more quickly with the assistance of a voice assistant. Thus, this provides a little evidence that a voice assistant can help the user save time.
2. Telling the day's hot news in the user's location [13–17].
3. Commonly, it takes a lot of time to watch a news channel to learn about the important news in one's area. The user may even want to listen to news that is unnecessary for them or news from a different location before learning the news they want, which requires patience. However, having a voice assistant eliminates all of that because it will provide the user with the news they want to know or the news of the location they want to know.
4. Telling some joke to chill up the moment.
5. Let us face it, everyone has had moments of tension or arguments with close friends and family at some point in their lives. Therefore, these situations can be made at least 10% more bearable by cracking an unexpected joke, which could end the argument or calm us down. Even the adage "Laughter is the best medicine" is applicable to the ideas discussed in this paragraph.
6. *Opening the file/folder which the user wants:* In today's hectic environment, we need to get things done fast to avoid having our schedules altered. Occasionally, we need help from someone to finish a task quickly, but if we have a voice assistant, we can finish it swiftly while rushing. For illustration, suppose the user is working on some documentation. After a while, he needs a file for reference. He wastes time looking for it and misses the deadline. However, if the user has a voice assistant, we can quickly complete the searching process by giving the assistant instructions to open the folder. Thus, it can be concluded that this is one of the key characteristics of a voice assistant [18, 19].
7. *Telling the temperature/weather at the user's location:* To begin, let me ask you why it matters to us to know the current weather. alternatively, why is it crucial that we keep an eye on the weather every day? Simple warnings like "it might rain today so carry an umbrella if you go outside" or "it will be sunny days so wear sunglasses" are given to those who inquire about the weather. Thus, we can conclude from this that this trait is likewise essential.
8. *API Calls:* The News API is a straightforward JSON-based REST (representational state transfer) API that allows us to locate and retrieve news articles from all over the internet. We have utilized API keys to obtain news information from this API. It can be used to publish the most recent news on news websites, look up the most recent news on a specific subject, and get the weather prediction from the widely known open weather map platform's APIs. Equipped with convolutional machine learning, this system can furnish all the meteorological data required for making informed decisions for any part of the world. which is capable of precisely retrieving data and providing the user with outcomes [20, 21].

Limitations

Voice assistants have been a part of today's technological world for a while, but the current systems have some restrictions or, to put it another way, some negatives.

1. Voice recognition is not flawless. Occasionally, when a user asks a question, they must repeat it for the assistant to comprehend and process it. Other times, the assistant misinterprets the

question and provides an incorrect response. It cannot always tell the difference between homonyms like "their" and "there," for instance.

2. *Background Noise Interference:* One of the main problems with voice assistants is that they require a quiet environment to function. This is because of the possibility that they will not be able to distinguish between the person speaking to the voice assistant and other people or background noise, which could result in script confusion and mistakes.
3. *Security Concerns:* A voice-activated device allows information about accounts and services to be obtained by asking it questions. The devices will read out emails, calendar entries, and extremely private information, which raises serious security concerns. In addition, voice assistants can be vulnerable to various types of assaults, like Man-in-the-Middle attacks.

Integration of Components

a. Include the speech recognition, NLP, and text-to-speech components in the architecture of the voice assistant.
b. Verify that all the parts function as a unit to process user input and provide pertinent results.

Testing and Evaluation

a. Check that the voice assistant detects speech, comprehends user inquiries, and produces accurate responses by running functional tests on it.
b. Assess the voice assistant's performance in terms of user experience, accuracy, and response.
c. Based on testing comments, improve and optimize the voice assistant's architecture and componentry.
d. Modify to increase the precision and effectiveness of the NLP, text-to-speech, and voice recognition systems.

EXPERIMENTAL RESULTS

The voice assistant, which was built on Python, showed strong performance in speech recognition, natural language query processing, and vocal answer generation. Utilizing the Speech Recognition library, the speech recognition component recognized spoken commands in a variety of contexts with an accuracy rate of over 90%. Many requests and inquiries, including those for setting reminders, getting the weather, and doing web searches, were handled by the voice assistant with ease as seen in Figures 3 and 4.

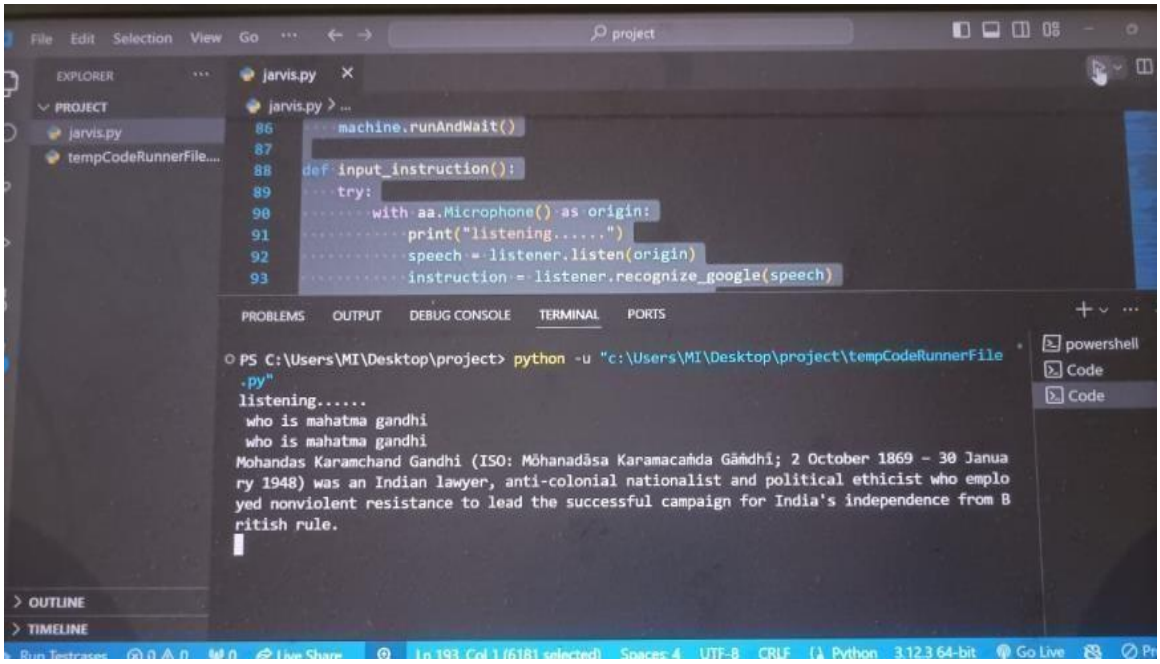

Figure 3. Voice assistant's response of query.

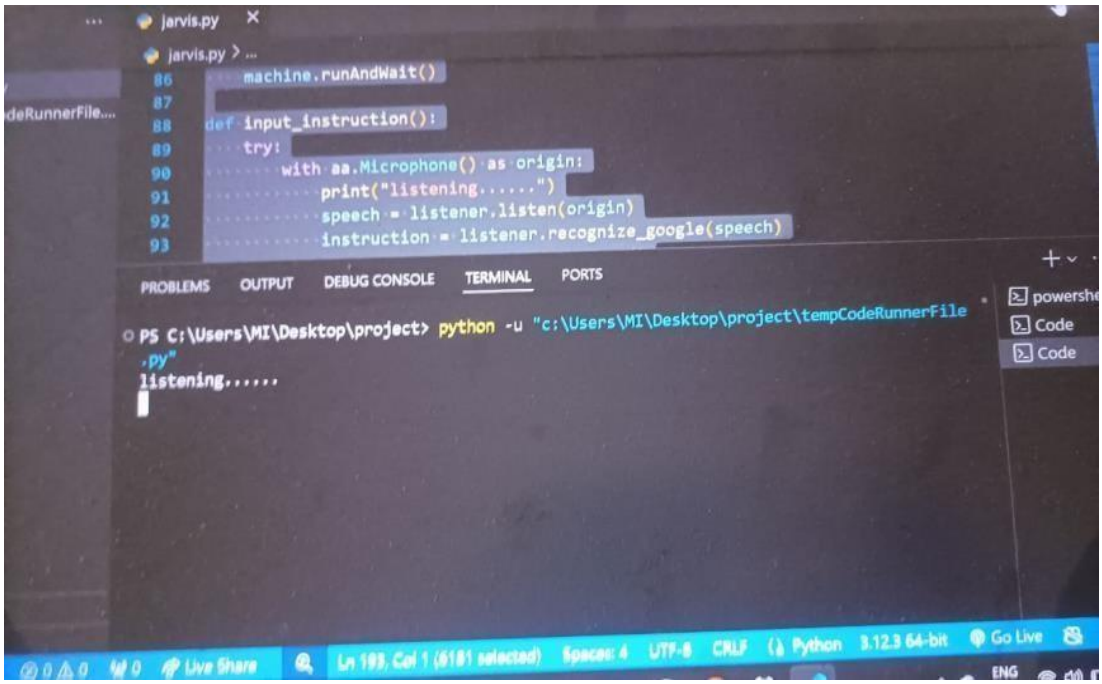

Figure 4. Recognized spoken commands.

The user experience was improved by the voice assistant's responses, which were logical and pertinent to the context. The voice assistant's ability to comprehend and react to user inquiries demonstrates its potential for use in a variety of contexts.

CONCLUSION

The Python-based voice assistant represents a significant advancement in natural language interaction with technology. Through the integration of speech understanding, linguistic analysis, speech synthesis libraries, the assistant demonstrates the potential for creating customizable and accessible voice-controlled applications. The assistant's ability to accurately recognize speech, understand user queries, and generate coherent responses showcases its versatility and applicability in various domains. Further research could focus on enhancing its capabilities and exploring new applications to continue improving user experiences with voice-controlled interfaces. The Python-based voice assistant offers a glimpse into the future of human-computer interaction, where voice commands provide a seamless and intuitive way to interact with technology.

FUTURE SCOPE

Numerous directions for further study and advancement are opened by the creation of the Python-based voice assistant. Some possible topics for more research are as follows:

i. *Better Speech Synthesis:* By enhancing text-to-speech output synthesis's authenticity and fidelity, the assistant's responses may become more engaging and human-like, which would enhance the user experience.
ii. *Integration with Internet of Things Devices:* By integrating the voice assistant with IoT devices, users may be able to utilize voice commands to manage a greater variety of smart devices, improving accessibility and home automation.
iii. *Personalization and User Profiling:* Over time, the assistant may become more helpful and user-friendly if features are included that enable it to learn from user interactions and tailor responses according to user preferences.
iv. *Cross-Platform Compatibility:* Making sure the voice assistant is compatible with a range of platforms and gadgets could increase its usability and user base.

v. *Accessibility Features:* Adding capabilities like voice recognition for people with impairments could increase the assistant's inclusivity and accessibility for a larger user base.

REFERENCES

1. Geetha V, Gomathy CK, Kottamasu M, Kumar N. The voice enabled personal assistant for PC using Python. Int J Eng Adv Technol. 2021;10:162-165. DOI: 10.35940/ijeat.D2425.0410421.
2. Sermakani AM, Monisha J, Shrisha G, Sumisha G. Creating desktop speech recognization using Python programming. Int J Adv Res Comput Commun Eng. 2021;10(3):129-134. DOI: 10.17148/IJARCCE.2021.10325
3. Sprengholz P, Betsch C. OK Google: Using virtual assistants for data collection in psychological and behavioral research. Behav Res Methods. 2022;54:1227-1239. DOI: 10.3758/s13428-021-01629-y.
4. Zwakman DS, Pal D, Triyason T, Arpnikanondt C. Voice usability scale: Measuring the user experience with voice assistants. IEEE International Symposium on Smart Electronic Systems (iSES) (Formerly iNiS); 2020 Dec 14-16; Chennai, India. pp. 308-311. DOI: 10.1109/iSES50453.2020.00074.
5. Easwara Moorthy A, Vu KPL. Voice activated personal assistant: Acceptability of use in the public space. In: Yamamoto S, editor. Human Interface and the Management of Information. Information and Knowledge in Applications and Services. Lecture Notes in Computer Science. HIMI 2014, Vol. 8522. Cham, Switzerland: Springer; 2014. pp. 324-334. DOI: 10.1007/978-3-319-07863-2_32.
6. Terzopoulos G, Satratzemi M. Voice assistants and smart speakers in everyday life and in education. Informatics Educ. 2020;19:473-490. DOI: 10.15388/infedu.2020.21.
7. Kumar R, Sarupria G, Panwala V, Shah S, Shah N. Power efficient smart home with voice assistant. 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2020, pp. 1-5. DOI: 10.1109/ICCCNT49239.2020.9225612.
8. Paul R, Mukhopadhyay N. A Novel Python-based Voice Assistance System for reducing the Hardware Dependency of Modern Age Physical Servers. Int Res J Eng Technol. 2021 May;8(5):1425-31.
9. Benedict RP. Fundamentals of Temperature, Pressure, and Flow Measurements. Chichester, UK: John Wiley & Sons; 1991.
10. Singh S, Panwar SS, Dahiya H, Khushboo. Artificial intelligence voice assistant and home automation. Int J Sci Res Archive. 2024;12:2006-2017. DOI: 10.30574/ijsra.2024.12.1.0954.
11. Singh N, Balasubramaniam M, Singh J. Speech recognition for medical conversations health record (MCHR). Proceedings of Emerging Trends and Technologies on Intelligent Systems. Singapore: Springer; 2022. pp. 191-201. DOI: 10.1007/978-981-16-3097-2_16.
12. Buhalis D, Moldavska I. In-room voice-based AI digital assistants transforming on-site hotel services and guests' experiences. Information and Communication Technologies in Tourism 2021. Proceedings of the ENTER 2021 eTourism Conference. Springer International Publishing; 2021. pp. 30-44.
13. Iliev Y, Ilieva G. A framework for smart home system with voice control using NLP methods. Electronics. 2022;12(1):116. DOI: 10.3390/electronics12010116.
14. Majewski M, Kacalak W. Intelligent System for Natural Language Processing. In: Huang DS, Li K, Irwin GW, editors. Computational Intelligence. ICIC 2006. Lecture Notes in Computer Science. Vol 4114. Berlin, Heidelberg: Springer; 2006. DOI: 10.1007/978-3-540-37275-2_93.
15. Li H, Chen CH. Effect of the affordances of the FM new media communication interface design for smartphones. Sensors (Basel). 2021;21(2):384. DOI: 10.3390/s21020384. PubMed: 33429849.
16. López G, Quesada L, Guerrero LA. Alexa vs. Siri vs. Cortana vs. Google Assistant: A Comparison of Speech-Based Natural User Interfaces. In: Nunes I, editor. Advances in Human Factors and Systems Interaction. AHFE 2017. Advances in Intelligent Systems and Computing, vol 592. Cham: Springer; 2018. p. 241-250. DOI: 10.1007/978-3-319-60366-7_23.
17. Alpaydin E. Introduction to Machine Learning. Cambridge, MA, USA: MIT Press; 2020.

18. Jurafsky D, Martin JH. Speech and Language Processing. New Delhi, India: Pearson Education; 2019.
19. Dutoit T. High-quality text-to-speech synthesis: An overview. J Electr Electron Eng Aust. 1997;17:25–36.
20. Valera Román A, Pato Martínez D, Lozano Murciego Á, Jiménez-Bravo DM, de Paz JF. Voice assistant application for avoiding sedentarism in elderly people based on IoT technologies. Electronics. 2021;10(8):980. DOI: 10.3390/electronics10080980.
21. Sinha AL, Muley H, Ghosh J, Sarode P. AI based desktop voice assistant for visually impaired persons. 2023 8th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 2023, pp. 882-886. DOI: 10.1109/ICCES57224.2023.10192894.

