Jarvis: AI-Powered Voice Genie

Dr. G. Kishor Kumar [1], Mr. A. Venkata Subbaiah Setty [2], Mrs. M. Lavanya [3], Mr.V. Dinesh Chowdary [4], Mr. T. Harsha Vardhan Naidu[5]
[1]Professor & Head Department of CSE&BS, RGMCET,
[2,3,4,5]Department of CSE&BS, RGMCET,

ABSTRACT

This research focuses on enhancing accessibility for individuals with disabilities who encounter challenges in operating conventional typing or clickable interfaces on Windows-based systems. To address this issue, we propose the development of an AI-driven voice assistant for desktop environments. The system incorporates advanced functionalities, including seamless Google search, automated email handling via Gmail integration, real-time news extraction, dynamic weather updates, and intuitive interactions with applications such as WhatsApp and YouTube. Built using Python, the assistant employs state-of-the-art voice recognition technology to enable a fluid and user-friendly experience. By redefining the user interface, the proposed solution ensures enhanced accessibility and ease of use for individuals with physical impairments. The assistant, named J.A.R.V.I.S. (Just A Rather Very Intelligent System), exemplifies the transformative potential of AI-driven personal assistants by demonstrating how voice-controlled systems can redefine human-computer interaction and improve digital accessibility.

Index terms - AI Voice Assistant, Accessibility, Speech Recognition, Human-Computer Interaction, Desktop Automation, Assistive Technology, Natural Language Processing, Gmail Integration, Real-Time News Extraction, Weather Forecasting, Voice-Controlled Interface, Python-Based Assistant, Smart Automation, Digital Accessibility, J.A.R.V.I.S.

INTRODUCTION

The rapid advancement of artificial intelligence (AI) and natural language processing (NLP) has led to the proliferation of voice assistants, significantly transforming human-computer interactions. These intelligent systems leverage speech recognition, NLP, and machine learning (ML) to process spoken language, interpret user intent, and execute corresponding commands. Prominent voice assistants, such as Apple's Siri, Google Assistant, and Amazon Alexa, have gained widespread adoption, primarily in mobile devices and smart home systems. However, despite the extensive deployment of these technologies, there remains a growing demand for their seamless integration into desktop computing environments to enhance user accessibility and productivity.

Conventional human-computer interaction methods, such as keyboard and mouse-based inputs, often pose usability constraints for individuals with physical disabilities or those requiring hands-free operation. Voice-assisted systems offer a viable alternative by enabling speech-based interaction, thereby improving accessibility and overall user experience. The development of a voice-controlled virtual assistant for desktop systems holds potential in automating routine computational tasks, facilitating information retrieval, and streamlining workflow efficiency.

This study presents the design and implementation of an AI-driven voice assistant optimized for desktop environments. The proposed system, developed using Python, integrates speech recognition technologies and automation frameworks to execute voice-controlled commands efficiently. The research aims to bridge the existing gap between mobile-based virtual assistants and desktop applications by offering a comprehensive solution capable of performing a wide range of functions, including web searches, email management, system control, and multimedia handling.

2.1 Evolution of Voice Assistants

The concept of voice assistants has evolved significantly over the years. Early implementations were limited to rule-based command recognition systems that converted speech into text without contextual understanding. With advancements in AI and NLP, modern voice assistants have achieved substantial improvements in accuracy, context awareness, and personalization. These systems operate in a structured manner involving three key processes:

1. **Speech-to-Text (STT):** Converts spoken language into text for further processing.
2. **Intent Recognition:** Analyze the input text to determine the user's intent and required action.
3. **Execution and Response:** Processes the command and generates an appropriate response or action.

By leveraging these processes, AI-driven voice assistants now facilitate various functionalities, including task scheduling, home automation, and knowledge retrieval, making them indispensable tools in modern digital ecosystems.

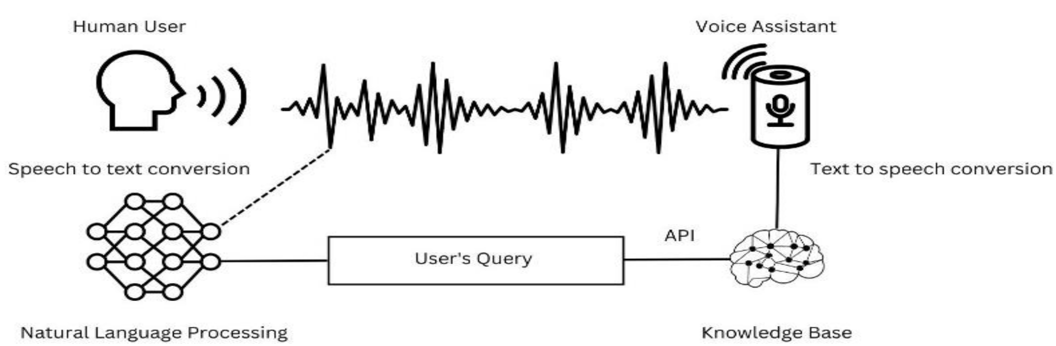


Fig 1 AI-driven voice assistants modern digital ecosystems

2.2 Applications and Significance

AI-powered voice assistants offer a broad spectrum of applications across multiple domains, including but not limited to:

- **Home Automation:** Facilitating control over smart devices such as lights, thermostats, and security systems.
- **Business and Enterprise Solutions:** Automating routine administrative tasks, managing emails, and optimizing workflow efficiency.

https://doi.org/10.5281/zenodo.15123556

- **Healthcare and Assistive Technology:** Providing hands-free assistance to individuals with disabilities, thereby improving accessibility.
- **Education and Research:** Enhancing digital learning through interactive voice-based query resolution and information retrieval.

Given the rising reliance on voice-assisted technologies in everyday tasks, developing a robust desktop-integrated voice assistant is crucial for enhancing productivity and ease of access across various operational environments.

2.3 Proposed System Overview

This research introduces **J.A.R.V.I.S. (Just A Rather Very Intelligent System),** an AI-based voice assistant designed explicitly for desktop computing. Unlike traditional voice assistants that primarily cater to mobile ecosystems, this system is tailored to function seamlessly within Windows-based environments. The proposed system is equipped with the following key features:

- **Voice-Controlled Web Searches:** Enabling hands-free browsing and information retrieval.
- **Email Automation via Gmail API:** Streamlining email communication through speech commands.
- **Real-Time News and Weather Updates:** Fetching and delivering dynamic updates.
- **Application and System Control:** Facilitating system navigation, application execution, and file management through voice input.
- **Integration with Messaging and Media Platforms:** Allowing interaction with services such as WhatsApp and YouTube via voice commands.

The system employs advanced speech recognition techniques, NLP models, and automation libraries to ensure accurate voice processing and efficient task execution. Furthermore, it is designed to function in both online and offline modes, making it a reliable and adaptive solution for users across diverse environments.

2.4 Research Significance and Future Scope

The integration of AI-driven voice assistants into desktop environments has the potential to redefine human-computer interaction by offering an intelligent, efficient, and hands-free computational experience. This research aims to provide a scalable and adaptive framework that can be further enhanced with deep learning models, multilingual support, and contextual awareness capabilities. Future developments may also explore enhanced personalization, improved response accuracy, and broader application integrations to expand the usability of AI-powered voice assistants in professional and personal computing environments.

This paper discusses the technical implementation, challenges, and prospective enhancements of the proposed system, emphasizing how AI and voice recognition technologies can significantly contribute to accessibility, automation, and user experience improvements in desktop computing.

LITERATURE SURVEY

Speech recognition has progressed remarkably throughout the years, experiencing multiple breakthroughs in technological developments. The incorporation of automatic speech recognition (ASR) for dictation, search capabilities, and voice commands has turned into a common feature in contemporary smartphones and wearable gadgets. The creation of a small, extensive vocabulary speech recognition system that functions effectively on mobile platforms with high precision and low latency is

 https://doi.org/10.5281/zenodo.15123556

a vital research domain. [1] This objective is achieved through the implementation of a Connectionist Temporal Classification (CTC)-based Long Short-Term Memory (LSTM) acoustic model, which predicts context-independent phonemes. The model undergoes compression to one-tenth of its original size by leveraging Singular Value Decomposition (SVD)-based compression techniques and quantization. Additionally, deep neural networks (DNNs) and real-time language model rescoring enhance performance, making speech recognition systems viable for modern smart devices. [2]

The ASR and search components facilitate speech recognition and retrieval tasks. In addition to these elements, a query parsing module is incorporated between ASR and search functions for enhanced accuracy and efficiency. [3] This integration introduces a set of methodologies aimed at improving automated voice search services for mobile users, who access these services through various portable devices. The voice search mechanism operates as a two-stage process, where string candidates generated by the ASR system are rescored to identify the most accurate match within a potentially vast, application-specific database. The study exemplifies how domain-specific knowledge sources can be combined with a domain-independent ASR system to enable efficient voice-based access to online search indices.

Prior to the advent of artificial intelligence (AI), technological upgrades relied heavily on human intervention to enhance functionality. However, contemporary AI-driven systems possess the capability to autonomously adapt to new tasks and solve problems without direct human involvement. This shift has led to significant improvements in automation, extending from mobile devices and personal computers to industrial automation processes. [4] The widespread adoption of voice assistants has streamlined human-computer interaction, boosting efficiency in various sectors.

In 2018, the Fourth World Conference on Smart Trends in Systems, Security, and Sustainability introduced strategies to improve human interaction with technology, aiming for greater efficiency and seamless integration. However, a challenge identified in this research was the occurrence of unintended activations, leading to delays in response. In 2020, the assessment of user experience quality for voice assistants introduced a method based on the User Experience Questionnaire Plus (UEQ+) framework, which evaluates various user experience dimensions. Despite its advancements, the study highlighted challenges related to speech recognition inaccuracies. Furthermore, in June 2020, the development of a domain-specific intelligent personal assistant with bilingual voice command processing by TENCON showcased a refined technology capable of automating multiple tasks. However, scalability remained a concern, particularly regarding homonyms—words with identical spellings but different meanings—which posed challenges in expanding the system's applicability to diverse contexts.

With continuous advancements in AI and natural language processing (NLP), voice assistants are evolving to address these limitations. The objective of this research is to develop a highly responsive, accurate, and adaptable voice assistant that mitigates existing challenges while enhancing accessibility, efficiency, and user interaction in desktop environments. By integrating AI-driven speech recognition, contextual understanding, and efficient execution of voice commands, the proposed system aims to redefine human-computer interaction, making technology more intuitive and inclusive for all users.[5]The article highlights the drawbacks of Generative AI (GenAI) for blind users, including accessibility issues, inaccuracies, and flawed mental models. Users struggle with unlabeled buttons, poor UI design, and difficulty verifying information. GenAI often generates hallucinations, reinforces ableist

https://doi.org/10.5281/zenodo.15123556

biases, and provides misleading responses. Blind users also develop oversimplified or incorrect mental models, leading to misplaced trust. The study underscores the need for more accessible, transparent, and reliable AI systems while balancing the benefits GenAI offers for content creation and information retrieval.[6] The article identifies key limitations of intelligent assistants, including definitional ambiguity, lack of standardization, and conceptual inconsistencies. The interchangeable use of terms like "intelligent personal assistant" and "virtual assistant" creates confusion, hindering theoretical and empirical research. Furthermore, many so-called intelligent assistants lack true AI capabilities, leading to misleading expectations. The study highlights the need for a clear framework distinguishing AI-enabled, interactive, and assistive technologies to enhance research, development, and public understanding.[7] The article highlights key drawbacks of the Smart Home Desktop Assistant, including limited user interaction, potential security risks, and reliance on an internet connection. Additionally, challenges such as inaccurate voice recognition, integration issues with IoT devices, and the need for continuous updates hinder its effectiveness.[8] The article highlights several limitations of the AI-based virtual mouse, including its dependency on camera quality, difficulty recognizing gestures in low light or complex backgrounds, and limited gesture set. Additionally, the system struggles with high-speed hand movements and lacks authentication features for user security.[9] The article outlines limitations of the personal voice assistant using computer vision, including dependence on high-quality audio input, difficulty in noisy environments, and lack of personalization. It also struggles with complex commands, requires continuous updates, and lacks integration with widely used digital services.[10] The article highlights several drawbacks of Jarvis AI, including challenges in understanding complex or ambiguous commands, reliance on high-quality speech input, and privacy concerns. It also struggles with personalization, lacks integration with some third-party services, and requires frequent updates to improve accuracy.

METHODOLOGY

The implementation of our AI-powered voice assistant is structured around robust speech synthesis and recognition technologies to enable seamless human-computer interaction. At the core of the system, **sapi5** and **pyttsx3** are utilized to provide text-to-speech conversion, ensuring offline functionality and compatibility across different Python versions. The Speech Application Programming Interface (SAPI), developed by Microsoft, allows the system to integrate speech recognition and synthesis capabilities within Windows applications, facilitating an efficient and responsive user experience.

The architecture of the proposed system is designed to continuously **listen for user commands** and process inputs dynamically. The assistant operates in an interactive loop, allowing users to set the listening duration based on their preferences. In scenarios where command recognition is unclear, the system is programmed to **prompt the user for clarification**, thereby improving accuracy and reducing erroneous responses.

Customization is a key feature of our system, offering users the ability to **choose between male and female voice outputs**, enhancing personalization and usability. Additionally, the assistant is equipped with a range of functionalities, including **weather updates, email management, Wikipedia search, application control, time queries, note-taking capabilities, and browser-based interactions** such as opening and closing websites like YouTube and Google.

The **modular and scalable** nature of the system allows for further enhancements and integration of advanced AI-driven features, ensuring adaptability to future technological advancements. The emphasis

 https://doi.org/10.5281/zenodo.15123556

on offline functionality and user-friendly customization makes this voice assistant a practical and accessible tool for a wide range of applications. As the system continues to evolve, improvements in natural language understanding, contextual awareness, and expanded command recognition will further enhance its performance and real-world applicability.

Voice assistants are entirely developed using programming languages, which process spoken commands and react based on the user's instructions. In this project, we utilized the Python Programming language to create the AI-powered Voice Assistant.

A user can express, “Play me a song” or “Open Youtube. com”, and the voice assistant will reply with the outcomes by playing the specified song or by launching the YouTube website. The Voice assistant remains on standby for a pause to indicate that users have completed their request, and subsequently, the voice assistant sends the user's request to its database for searching the requested item.

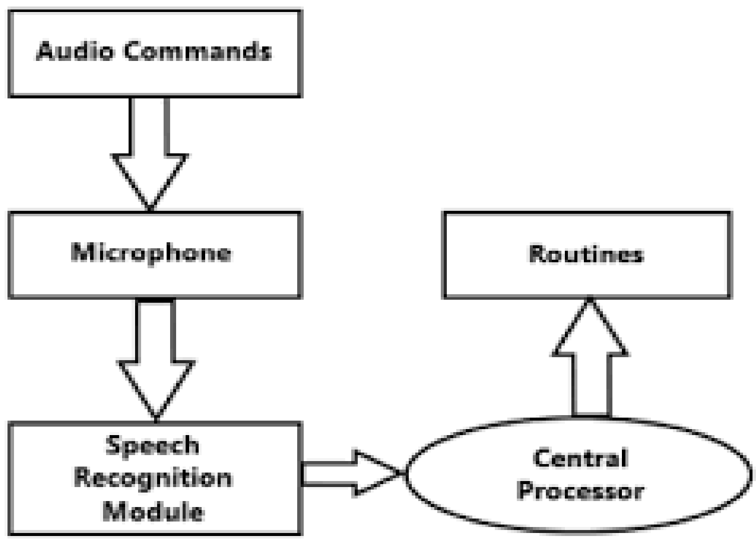


Fig 2 Control Flow Diagram

• The user's request is divided into distinct commands to ensure our voice assistant can comprehend it.
• After being included in the commands list, our request is looked up and matched against other requests.
• The commands list subsequently transmits these commands back to the Voice assistant.
• When the voice assistant gets those commands, it then knows the next steps to take.
• The voice assistant may pose a question if the request is unclear enough to process, in other terms, to confirm it understands what we want to receive.
• If it believes it has sufficient understanding to process it, the voice assistant will carry out the task that the user has requested.

 https://doi.org/10.5281/zenodo.15123556

System Architecture:

The voice assistant system consists of the following components:

- **Speech Recognition Module**: Captures user input through a microphone and converts it into text.
- **Natural Language Processing (NLP) Engine**: Analyses user commands and determines appropriate responses.
- **Text-to-Speech (TTS) System**: Utilizes the gTTS engine to convert text responses into audible speech.
- **Task Execution Module**: Implements predefined functionalities, such as web search, media control, email handling, and security alerts.
- **User Personalization and Session Memory**: Allows the assistant to remember names, preferences, and frequently used commands for improved interaction.

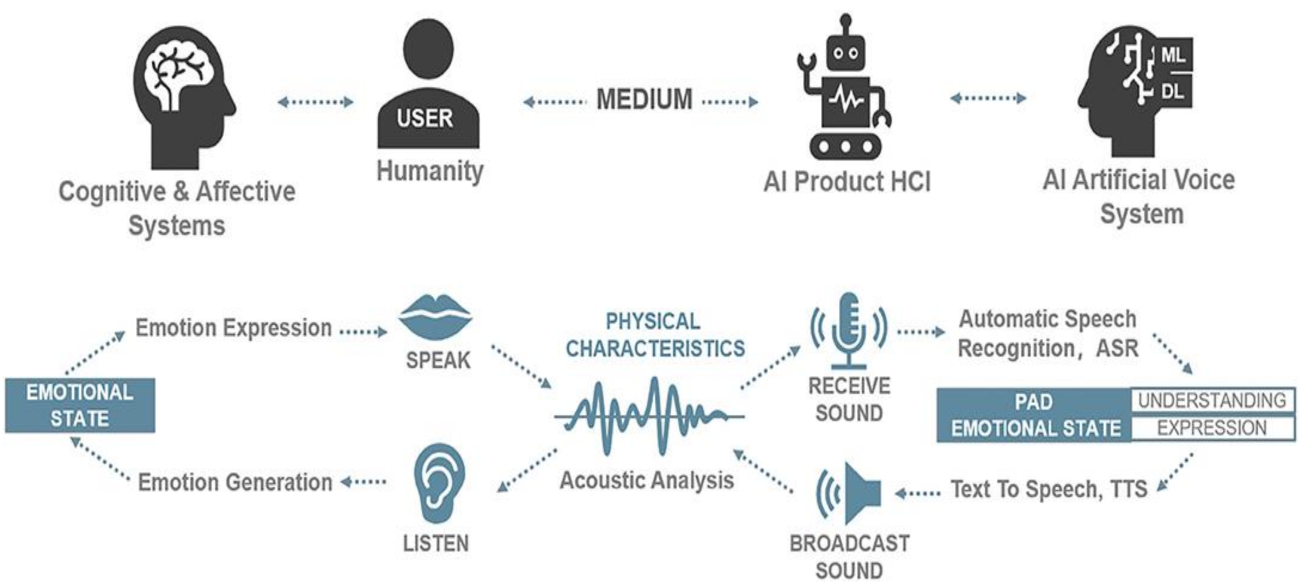


Fig 3 System Architecture

Algorithms

5.1 Speech Recognition Algorithms

5.1.1 Speech Input Processing

Algorithm: Google Speech Recognition (via SpeechRecognition library)

The function captures real-time audio input using sr.Microphone(), enabling voice-based interactions. To reduce background noise and improve recognition, r.pause_ threshold = 1 allows brief pauses without terminating input. The speech recognition model is configured for English (Indian dialect) using language='en-in', ensuring better accuracy for regional accents.

5.1.2 Speech Recognition Method

 https://doi.org/10.5281/zenodo.15123556

5.1.3 Core Recognition Techniques

The function dynamically adjusts the noise threshold to filter out background disturbances for clearer audio input. It performs real-time speech-to-text conversion while incorporating context-aware recognition for better accuracy. The system processes spoken commands efficiently, adapting to different speech patterns. If speech is unrecognized, it implements a fallback mechanism to prevent errors and ensure smooth execution.

5.2. Text-to-Speech (TTS) Algorithms

5.2.1 SAPI5 Text-to-Speech Implementation

The pyttsx3 library is used for text-to-speech conversion, enabling voice-based responses. The speech engine is initialized using sapi5, a Microsoft speech API, ensuring smooth synthesis. Available voices are retrieved with engine.getProperty ('voices'), and a specific voice is selected using voices[0].id. The speaking rate is set to 150 for clear and natural-sounding speech output.

5.2.2 Speech Synthesis Method

The speak(audio) function utilizes the text-to-speech engine to convert text into spoken output. It takes an input string (audio) and passes it to engine.say(), which prepares the speech synthesis. The engine.runAndWait() command ensures the speech is fully delivered before proceeding. This function enables voice-based responses in applications, enhancing user interaction.

5.2.3 Key TTS Parameters

The system supports multiple voice options, allowing users to select their preferred voice for speech output. The speech rate is adjustable, with a default setting of 150 words per minute to ensure clear pronunciation. It generates audio output synchronously, ensuring that spoken text is fully delivered before proceeding. This enhances the user experience by providing smooth and natural voice responses.

5.3 Advanced Recognition Techniques

5.3.1 Context-Aware Command Processing

The algorithm uses keyword-based command matching to identify user instructions efficiently. It first converts the input query to lowercase, ensuring uniformity in text processing. Multiple conditional checks are performed to compare the query against predefined commands. Additionally, the system interprets commands contextually, enhancing accuracy and responsiveness in executing user requests.

5.3.2 Error Resilience

The system incorporates graceful error handling to manage unexpected issues without crashing. It implements multiple retry mechanisms to improve command recognition and execution reliability. If errors persist, the program falls back to default responses, ensuring a smooth user experience. These features enhance robustness and adaptability in handling diverse inputs.

5.4. Performance Optimization Strategies

5.4.1 Computational Efficiency

The system is designed with minimal computational overhead, ensuring smooth performance without excessive resource consumption. It enables low-latency speech recognition, allowing real-time responses with minimal delays. Efficient memory management optimizes resource usage, preventing unnecessary

 https://doi.org/10.5281/zenodo.15123556

load on the system. These features enhance overall speed, responsiveness, and reliability in processing voice commands.

5.4.2 Scalability Considerations

- Modular algorithm design
- Easy integration of additional voice commands
- Platform-independent implementation

The proposed voice-controlled assistant demonstrates a robust implementation of speech recognition and synthesis algorithms, providing a foundation for intelligent voice interaction systems.

EXPERIMENTAL RESULTS

6.1. Authentication & Access Control

- Face Recognition Authentication to Access Assistant
- Password/PIN-based Authentication
- Voice Recognition for Secure Access

6.2. Basic System Commands

- Welcome and Wishing
- What is the Time Command
- Check Date Command
- System Volume Control (Increase/Decrease/Mute)
- Open Applications (e.g., Open Notepad, Open Chrome)
- Shutdown, Restart, or Sleep System
- Lock Screen Command

6.3. Information Retrieval

- Google Search
- Wikipedia Search
- Showing Weather Report for a Particular Place
- Fetching News Updates
- Checking CPU & Battery Status

6.4. Communication & Messaging

- Sending Email
- Reading Emails
- Sending WhatsApp Messages
- Sending SMS

https://doi.org/10.5281/zenodo.15123556

6.5. Entertainment & Media Control

- Playing Music (Play, Pause, Stop, Next, Previous)
- Opening YouTube & Playing Specific Videos
- Opening Netflix, Prime Video, or Other Streaming Services
- Telling Jokes or Fun Facts
- Reading E-books or PDFs

6.6. Smart Home & IoT (if integrated)

- Turning On/Off Smart Lights
- Adjusting Room Temperature (If Smart Devices Are Connected)
- Controlling Smart Appliances

6.7. Productivity & Office Work

- Setting Alarms and Reminders
- Creating To-Do Lists
- Scheduling Meetings
- Opening and Editing Documents
- Converting Text to Speech

6.8. Programming & Development Assistance

- Running Code in Python/Java (if integrated)
- Searching for Coding Errors and Fixes
- Fetching Documentation for Programming Languages

6.9. Social Media & Web Navigation

- Opening Facebook, Twitter, LinkedIn, etc.
- Posting Updates to Social Media
- Fetching Trending Topics

6.10. Utility & Miscellaneous Commands

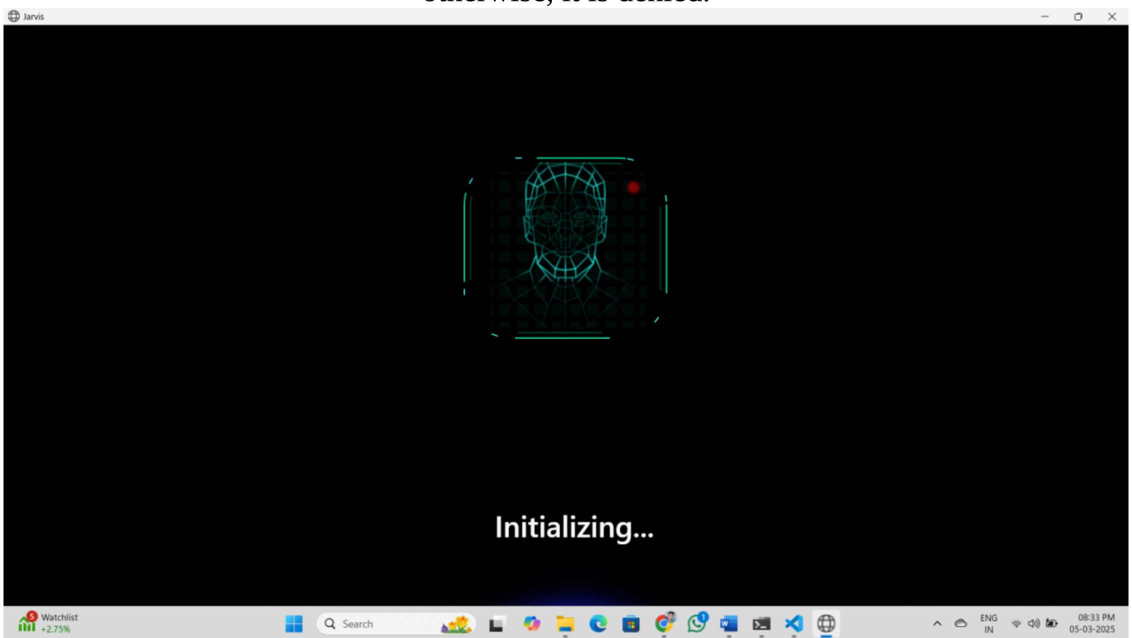
- Translating Languages
- Currency Conversion
- Unit Conversion (e.g., cm to inches)
- Checking Stock Market Prices
- Providing Health Tips

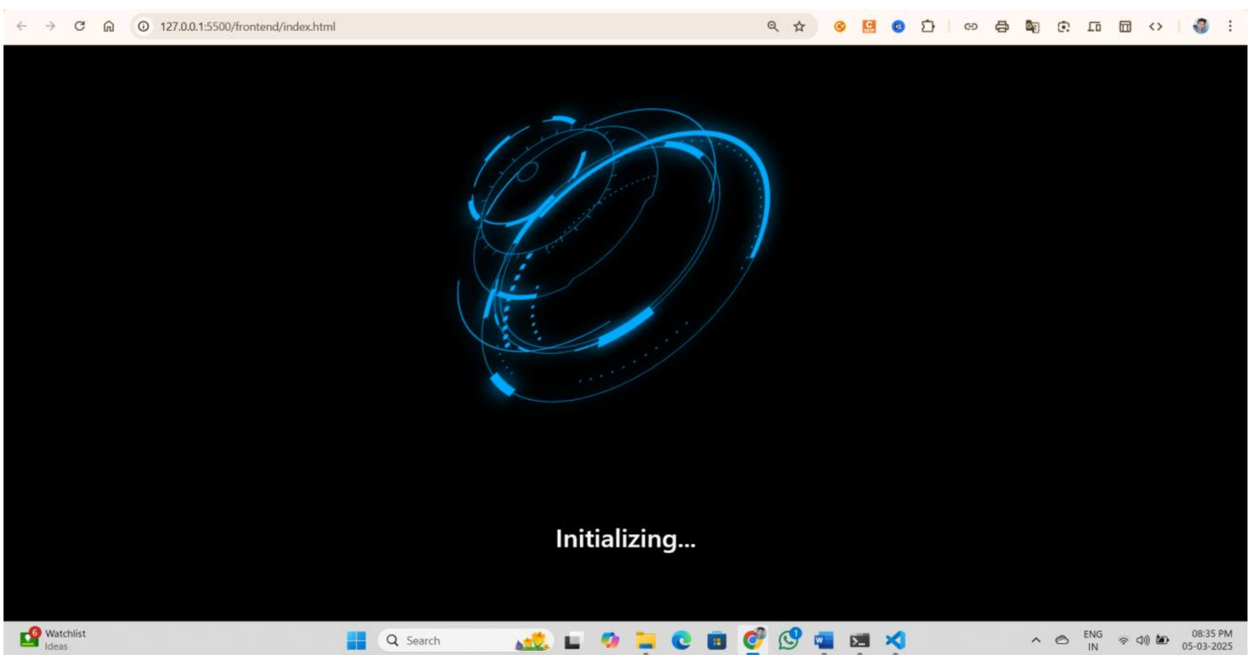
7. EXECUTION

https://doi.org/10.5281/zenodo.15123556

7.1. Face Recognition Authentication to Access Assistant

Explanation: This screenshot captures the face recognition authentication process. The assistant verifies the user's identity before granting access. If the face matches the stored data, access is approved; otherwise, it is denied.



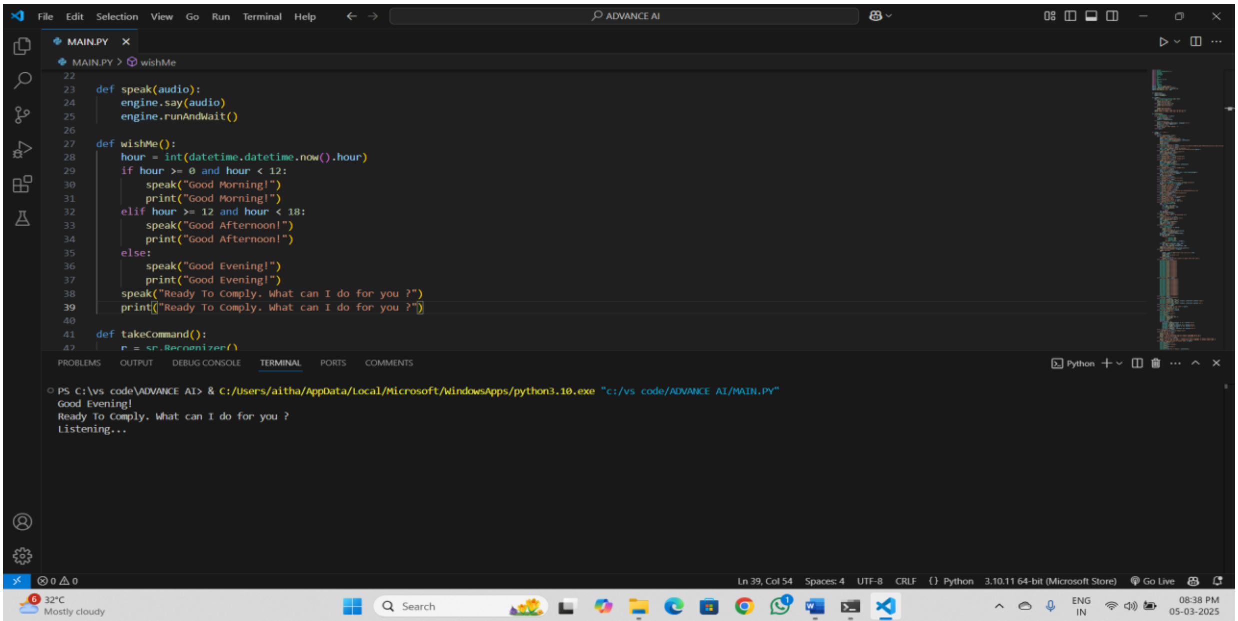


https://doi.org/10.5281/zenodo.15123556

 Volume 94, No. 2, 2025

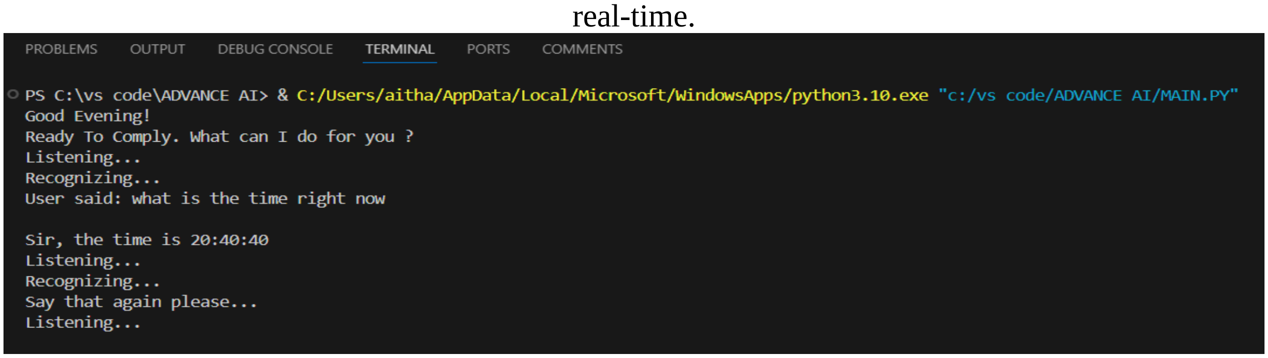
7.2. Welcome and Wishing

Explanation: This screenshot shows the assistant greeting the user based on the time of the day. The assistant dynamically wishes "Good Morning," "Good Afternoon," or "Good Evening" after successful authentication.



7.3. What is the Time Command

Explanation: This screenshot displays the assistant responding to the "What is the time?" command by fetching and announcing the current system time. The response format includes hours and minutes in real-time.

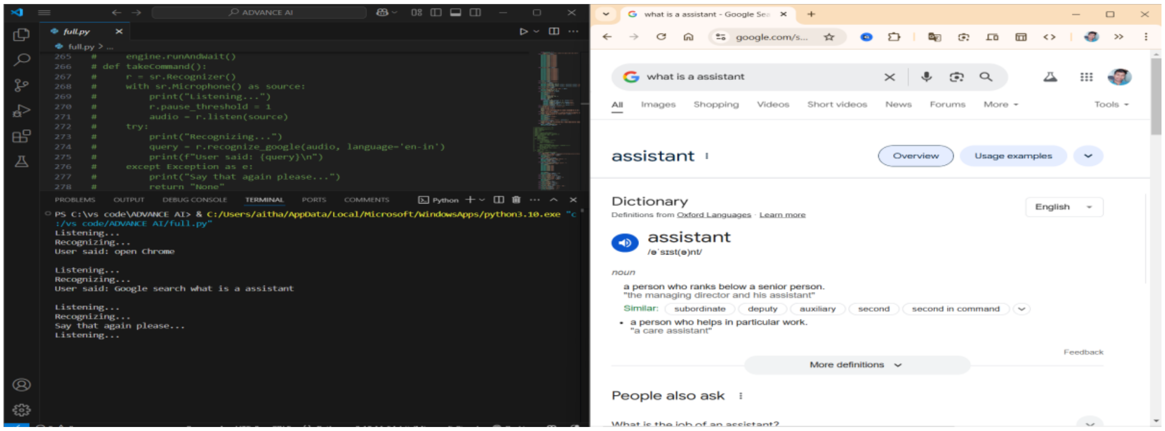


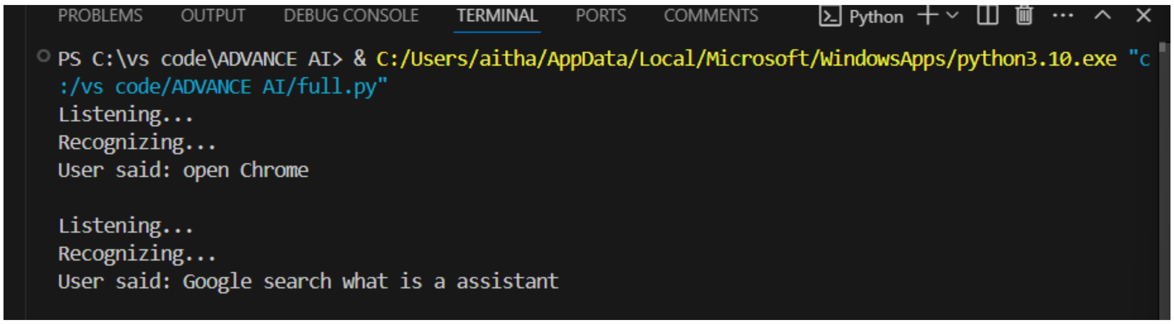
7.4. Google Search

Explanation: This screenshot demonstrates the assistant performing a Google search based on a user's query. The assistant processes the command, retrieves relevant search results, and either provides a verbal response or opens the browser with search results.

 https://doi.org/10.5281/zenodo.15123556

 Volume 94, No. 2, 2025





7.5. Showing Weather Report for a Particular Place

Explanation: This showcases the assistant fetching and displaying the weather report for a specified location. The response includes temperature, humidity, and weather conditions, providing real-time weather updates to the user.

7.6. Sending Email

Explanation: This functionality will the assistant composing and sending an email as per the user's command. It confirms the successful delivery of the email and may also display a sent message in the email client.

CONCLUSION

The integration of AI-powered voice assistants has revolutionized human-computer interaction by enabling hands-free operation and enhancing accessibility across various domains. In our project, we have incorporated advanced functionalities that extend beyond traditional voice assistants, making it more efficient and user-friendly. The widespread adoption of voice assistants in industries such as healthcare, research laboratories, business environments, and smart home systems underscores their significance in modern-day applications.

A key advantage of voice assistants lies in their ability to bridge the digital divide by providing accessibility to individuals with limited literacy or physical constraints. In laboratory environments, where professionals are required to wear protective gear, voice-based interaction facilitates seamless access to information without manual intervention. Similarly, in educational and domestic applications,

 https://doi.org/10.5281/zenodo.15123556

voice assistants contribute to increased efficiency and convenience.

The continuous advancements in artificial intelligence, natural language processing, and machine learning have led to significant improvements in the accuracy, responsiveness, and contextual understanding of voice assistants. Over the past few years, these systems have evolved to offer enhanced interaction capabilities, personalized experiences, and improved integration with IoT devices. Moreover, the growing reliance on voice assistants among younger demographics highlights their increasing acceptance as an essential tool in daily life.

As technology progresses, future developments in voice assistant systems will focus on enhancing conversational intelligence, multi-lingual support, privacy and security measures, and real-time adaptability to user preferences. With ongoing research and innovation, AI-driven voice assistants are expected to play an even greater role in shaping human-computer interactions, contributing to the digital transformation across multiple sectors.

FUTURE SCOPE

The rapid advancements in artificial intelligence and voice recognition technologies present significant opportunities for the continued evolution of AI-powered voice assistants. Future developments will focus on enhancing accessibility, personalization, and adaptability, making these systems more inclusive and efficient across diverse user demographics and application domains.

One of the key areas of improvement is **expanding accessibility features** to support individuals with various disabilities. By integrating advanced speech synthesis and recognition models, voice assistants can cater to users with speech impairments, cognitive disabilities, or motor function limitations, ensuring seamless interaction with digital systems.

The incorporation of **machine learning algorithms** will further refine command recognition by improving contextual understanding, reducing error rates, and enabling adaptive learning based on user behavior. This will lead to a more intuitive and personalized user experience. Additionally, **multilingual support** will enhance the accessibility of voice assistants across global markets, allowing seamless interaction in multiple languages and dialects through sophisticated natural language processing (NLP) techniques.

To address reliability concerns in areas with limited internet connectivity, **developing offline functionality** for core features will be a crucial advancement. By leveraging on-device AI processing, voice assistants can execute essential commands, retrieve stored data, and perform basic tasks without requiring continuous network access.

Furthermore, providing users with the ability to **customize command vocabulary** will enhance adaptability, allowing individuals to define personalized commands tailored to specific needs and preferences. This will be particularly beneficial in specialized domains such as healthcare, education, and industrial automation, where unique terminologies and workflows require flexible voice interactions.

https://doi.org/10.5281/zenodo.15123556

As research and development in AI and NLP keep advancing, the future of voice assistants will feature greater intelligence, contextual understanding, and innovations focused on the user. These developments will improve the effectiveness of digital interactions as well as support a more inclusive and accessible technological environment.

REFERENCES

[1] DO'SHAUGHNESSY,SENIOR MEMBER, IEEE,"Interacting With Computers by Voice: Automatic Speech Recognition and Synthesis proceedings of THE IEEE,VOL.91,NO.9, Gregorian calendar month 2003

[2] Nil Goksel-Canbek Mehmet Emin Mutlu,"On the track of Artificial Intelligence: Learning with intelligent Personal Assistant" International Journal of Human Sciences.

[3] VINAY SAGAR, KUSUMASM, "Home Automation Using net of Things", June-2015, IRJET, e- ISSN:2395-0056.

[4] "Speech recognition with flat direct models," IEEE Journal of elect Topics in Signal Processings, 2010

[5] Adnin, R., & Das, M. (2024, October). " I look at it as the king of knowledge": How Blind People Use and Understand Generative AI Tools. In *Proceedings of the 26th International ACM SIGACCESS Conference on Computers and Accessibility* (pp. 1-14).

[6] Shaikh, S. J. (2023). Artificially intelligent, interactive, and assistive machines: A definitional framework for intelligent assistants. *International Journal of Human–Computer Interaction*, *39*(4), 776-789.

[7] Ramachandran, S., Saha, B., Khandelwal, P., Rajesh, M., & Arul, R. (2023, December). Design and Development of Smart Home Desktop Assistant. In *2023 Innovations in Power and Advanced Computing Technologies (i-PACT)* (pp. 1-5). IEEE.

[8] Khandagale, A., Thakkar, N., Patil, S., Jadhav, V., & Nehete, C. (2023, May). Jarvis-AI based virtual mouse. In *2023 International Conference on Signal Processing, Computation, Electronics, Power and Telecommunication (IConSCEPT)* (pp. 1-5). IEEE.

[9] Jagan, A., Pasha, M. G., Nandini, D., Susanna, H., & Parvathi, D. N. (2023, November). Personal Voice Assistant Using Computer Vision. In *2023 International Conference on Research Methodologies in Knowledge Management, Artificial Intelligence and Telecommunication Engineering (RMKMATE)* (pp. 1-8). IEEE.

[10] Gupta, M., Kumar, R., & Sardalia, H. (2023, May). Voice assistant technology: The case of jarvis ai. In *2023 4th International Conference for Emerging Technology (INCET)* (pp. 1-5). IEEE

https://doi.org/10.5281/zenodo.15123556