IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 13, April 2026
Impact Factor: 8.2

AIVA: AI-Based Desktop Voice Assistant

Prof. Vikas Gaikwad[1], Shraddha Takmoge[2], Saloni C.[3], Akanksha Uchale[4], Sharvaree Jakate[5]
Professor, Department of Artificial Intelligence & Data Science[1]
Students, Department of Artificial Intelligence & Data Science[2-5]
Shree Ramchandra College of Engineering, Pune, India

Abstract: *This paper presents AIVA, a Python-based intelligent personal assistant engineered to automate complex desktop operations and streamline system interactions. Unlike standard virtual assistants, AIVA focuses on deep integration with local environments, enabling hands-free control over applications such as Microsoft Word, PowerPoint, and WhatsApp. A key focus of this architecture is strict modularity and portability; the system dynamically resolves directories rather than relying on static, hardcoded file paths, allowing for seamless deployment and sharing across different local machines. Furthermore, the system prioritizes an interactive user experience by integrating robust text-to-speech capabilities directly into the front-end user interface, ensuring system feedback is delivered vocally rather than through standard console outputs.*

Keywords: Artificial Intelligence, Natural Language Processing, Speech Recognition, Text-to-Speech, Speech-to-Text, Desktop Voice Assistant

I. INTRODUCTION

As digital workflows become increasingly complex, there is a growing need for localized, highly customizable automation tools. While commercial voice assistants excel at web-based queries, they often lack the deep system permissions required to manipulate local desktop applications effectively. AIVA was developed to bridge this gap. By utilizing a combination of Application Programming Interfaces (APIs) and user interface (UI) bypass methods, AIVA allows users to execute multi-step application commands using natural language. This paper details the architecture, development process, and modular design principles behind the AIVA system.

II. PROBLEM STATEMENT

Popular voice assistants are primarily mobile-oriented and cloud-based, leaving a gap for localized desktop solutions. Currently, desktop users lack efficient, hands-free tools to automate routine system operations, which slows down productivity and limits accessibility for differently-abled individuals. Therefore, an AI-based desktop voice assistant is needed to interpret spoken commands for local tasks, ultimately reducing manual effort and saving time.

III. OBJECTIVE

- To develop a desktop voice assistant that can understand and respond to voice commands.
- To integrate multiple functionalities like web browsing, file management, and application control.
- To reduce manual effort by automating routine tasks through voice interaction.
- To enhance user experience using AI-driven speech recognition and NLP techniques.

IV. SCOPE

The system focuses on delivering a seamless desktop experience through three primary capabilities:

- System Automation: Executes basic system-level operations, enabling users to open apps, close programs, and search files hands-free.
- Information Retrieval: Acts as a direct line to the internet, providing quick information retrieval directly from the web.

DOI: 10.48175/IJARSCT-33954

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 13, April 2026 Impact Factor: 8.2

• API Integration: Extends core functionalities by connecting with third-party APIs to provide live weather updates, news, and targeted services.

The core of the system is an intelligent, completely voice-enabled interface capable of bridging the gap between a user and their local desktop environment. Users can interact with the system through simple voice commands, significantly reducing the dependency on manual typing and navigation.

The system will also provide hands-free assistance for routine computer operations, such as automating emails and playing music or videos through voice commands. This will help enhance daily productivity and reduce the physical effort typically required to perform repetitive desktop tasks.

V. LITERATURE SURVEY

TABLE I: LITERATURE SURVEY

Sr No.	Title	Author	Year	Methodology Used	Conclusion
1.	"AI-based Desktop Voice Assistant"	Pankaj Kunekar, Ajinkya Deshmukh, Sachin Gajalwad, Aniket Bichare, Kiran Gunjal, Shubham Hingade	2023 International Conference on Nascent Technologies in Engineering (ICNTE 2023)	The system uses Python, Google's speech recognition, and NLP to match voice input to commands. It utilizes APIs for weather and news, alongside reading, writing, and audio visualization modules.	The desktop application successfully performs basic tasks like checking weather and opening sites without cloud services. Future versions could integrate machine learning and IoT to control nearby devices.
2.	"Voice Assistant Using Python"	Nivedita Singh, Dr. Diwakar Yagyasen, Mr. Surya Vikram Singh, Gaurav Kumar, Harshit Agrawal	July 2021| IJIRT | Volume 8 Issue 2	Created a desktop assistant for Windows and Linux utilizing Python libraries and machine learning model.. It translates voice to English phrases and redirects tasks to a Linux server via HTTP.	The application performs basic tasks like weather updates and Wikipedia searches. Future updates will incorporate machine learning and IoT to control nearby devices
3.	Research Paper on Desktop Voice Assistant	Vishal Kumar Dhanraj, Lokeshkriplani, Semal Mahajan	International Journal of Research in Engineering and Science (IJRES) , Volume 10 Issue 2 ‖ 2022 ‖ PP. 15-20	Built a Python-based virtual assistant using Google Speech API for recognition and gTTS for text-to- speech output. It integrates modules like WolframAlpha, Wikipedia, OS, and Pyjokes to execute system calls.	The developed voice assistant performs tasks without error and filters out ambient noise. Virtual personal assistants are considered more portable, loyal, and available than human assistants.

VI. METHODOLOGY

The development of the AIVA system follows a structured approach combining NLP and desktop automation to provide an efficient, hands-free user experience.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 13, April 2026
Impact Factor: 8.2

The process involves:

A. Requirement Analysis:

Identify the core needs of desktop users, focusing on task automation, quick information retrieval, and ensuring the software remains highly portable across different environments.

B. System Design:

Construct a flexible pipeline utilizing Faster-Whisper for Speech-to-Text (STT) , a local LLM for understanding commands , and an execution engine for system control.

C. API & Data Integration:

Connect external APIs (such as OpenWeather and NewsAPI) to fetch live information. The system dynamically resolves local directories rather than relying on fixed paths, ensuring it works seamlessly when deployed on any machine.

D. Intent Processing:

Configure the Command Processing Engine using Ollama and LangChain. This layer analyzes the transcribed text to accurately determine the user's intended action.

E. Implementation:

Build a visual dashboard using PyQt6. To maintain a true voice-assistant experience, traditional text outputs are bypassed, and all system responses and confirmations are delivered vocally using Edge TTS.

F. Testing & Validation:

Perform real-world testing to verify low-latency transcription, accurate desktop application control (such as launching browsers, Spotify, or WhatsApp), and reliable overall system performance.

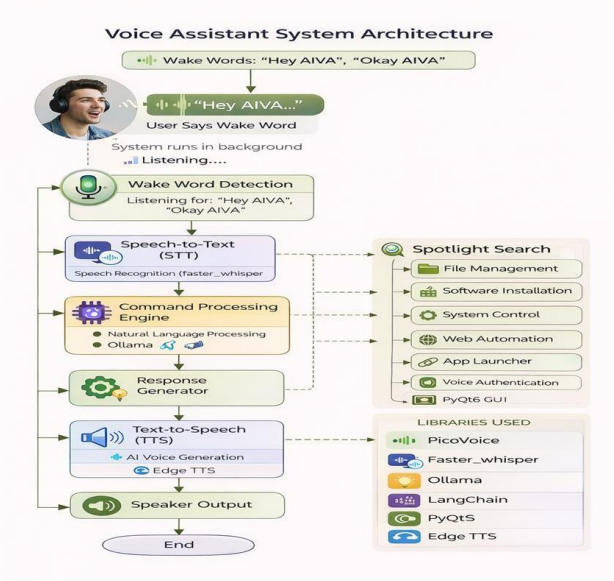

Figure 1. System Functions

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 13, April 2026
Impact Factor: 8.2

VII. MODELING & ANALYSIS

The AIVA system incorporates several key processing and analytical components to ensure accurate voice recognition, intelligent intent deduction, and seamless task execution.

A. Acoustic Processing and Speech Recognition

1) Purpose: To capture and accurately convert user voice commands into machine-readable text in real-time.
2) Technique: The system employs local acoustic modeling to minimize dependency on high-latency cloud services.
3) Transcription Engine: The faster-whisper library is utilized for high-speed, local speech-to-text (STT) transcription. This ensures that environmental audio is rapidly transcribed into text strings, providing a highly responsive initial input layer for the assistant.

B. Natural Language Processing & Intent Classification

1) Purpose: To intelligently deduce the user's core objective from the transcribed text command.
2) Technique: The system utilizes local Large Language Models (LLMs) orchestrated through the LangChain framework.
3) Semantic Analysis: Instead of relying on rigid, hardcoded keyword matching—which frequently fails with natural human speech variations—the NLP engine, powered by Ollama, analyzes the semantic context of the phrase. It classifies the intent (e.g., "media playback", "API fetch") and extracts relevant parameters to securely pass to the execution engine.

C. Execution Routing and Dynamic Resolution

1) Purpose: To map the analyzed intent to the correct functional automation script without encountering environment or pathing errors.
2) Technique: The central command processing engine acts as a decision router utilizing dynamic directory mapping and API integration.
3) Automation Execution: For web-based tasks, the system triggers Selenium for robust browser control. For local system applications, it utilizes dynamic path resolution logic, analyzing system variables at runtime rather than relying on static, hardcoded directories. This ensures the software remains highly portable and functional across different desktop environments without the instability of basic "type at cursor" automation.

D. Vocal Synthesis and Feedback Generation

1) Purpose: To provide natural, hands-free confirmation of task execution or query results.
2) Technique: A voice-first interface design that strictly bypasses standard terminal text outputs for user-facing communication.
3) Text-to-Speech Integration: Once the execution routing retrieves data (e.g., fetching a JSON response from the OpenWeatherMap API), the raw data is formatted into a conversational string. The Edge TTS library analyzes this text and generates a natural-sounding vocal output, ensuring the user receives immediate auditory feedback for a complete hands-free loop.

VIII. SYSTEM DESIGN

The system is built on a modular architecture with the following components: The system is built on a modular architecture with the following components:

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 13, April 2026
Impact Factor: 8.2

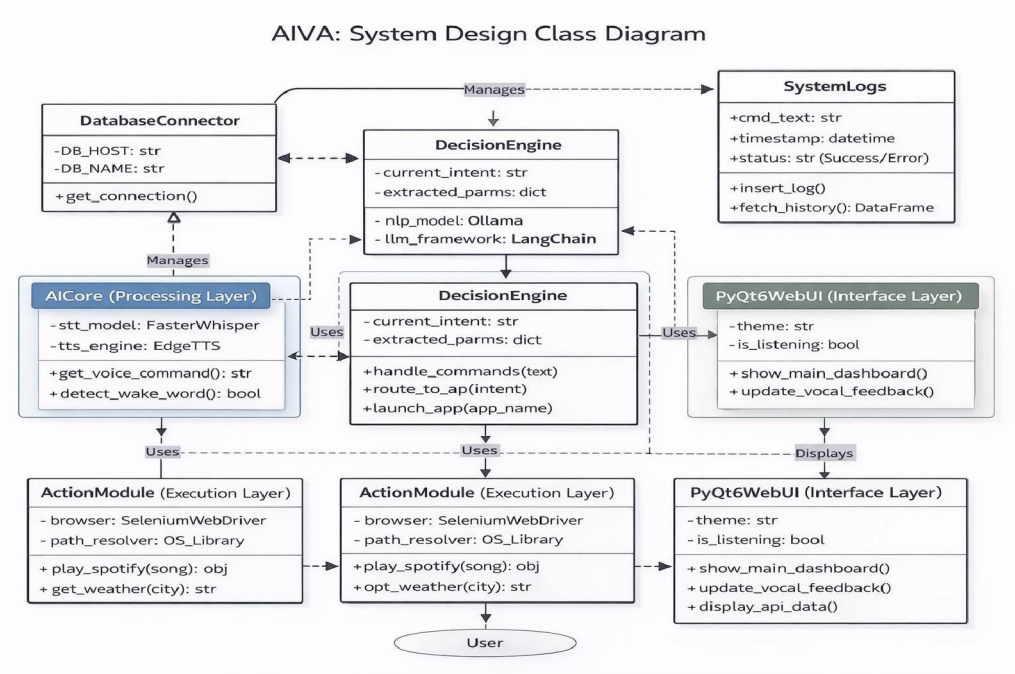

Figure 2. System Design

A. UI Layer:
PyQt6 dashboard with a background listener for wake-word detection.
B. Processing Layer:
Converts audio to text via Faster-Whisper and analyzes intent using Ollama/LangChain.
C. Execution Layer:
Automates web tasks via Selenium and local tasks using dynamic OS path resolution.
D. Feedback Layer:
Delivers natural-sounding vocal responses using Edge TTS.

IX. RESULT & DISCUSSION

The implementation of AIVA demonstrates a highly responsive system capable of automating complex desktop operations through a voice-first interface. The system successfully bridged the gap for localized automation, saving time and reducing the manual effort typically required for repetitive desktop navigation.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 13, April 2026
Impact Factor: 8.2

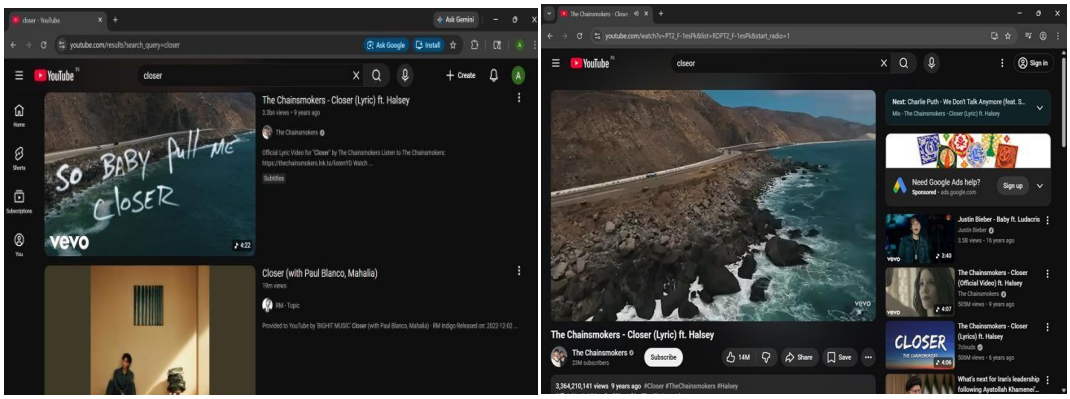

Figure 3. Result – Playing song on YouTube

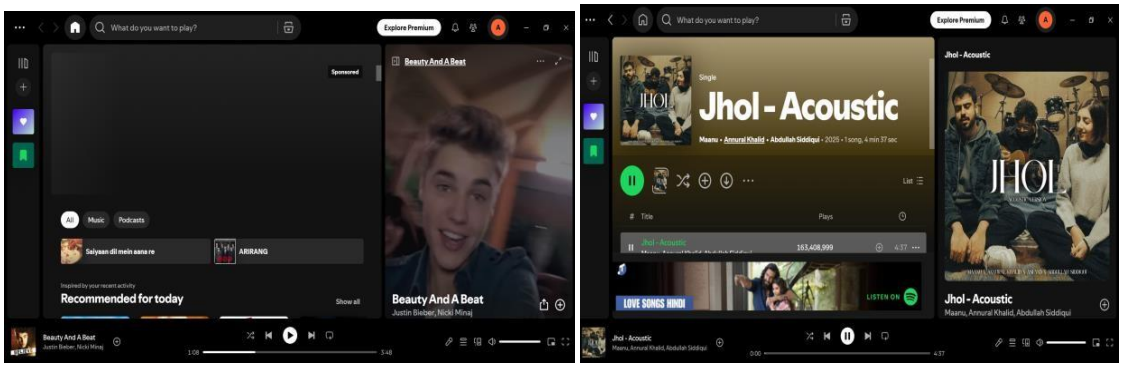

Figure 4. Result – Playing song on Spotify

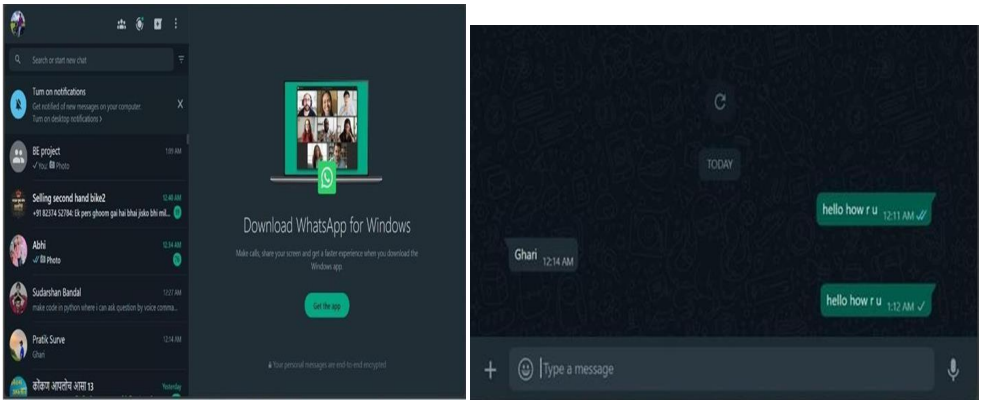

Figure 5. Result – Working on Whatsapp

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 13, April 2026
Impact Factor: 8.2

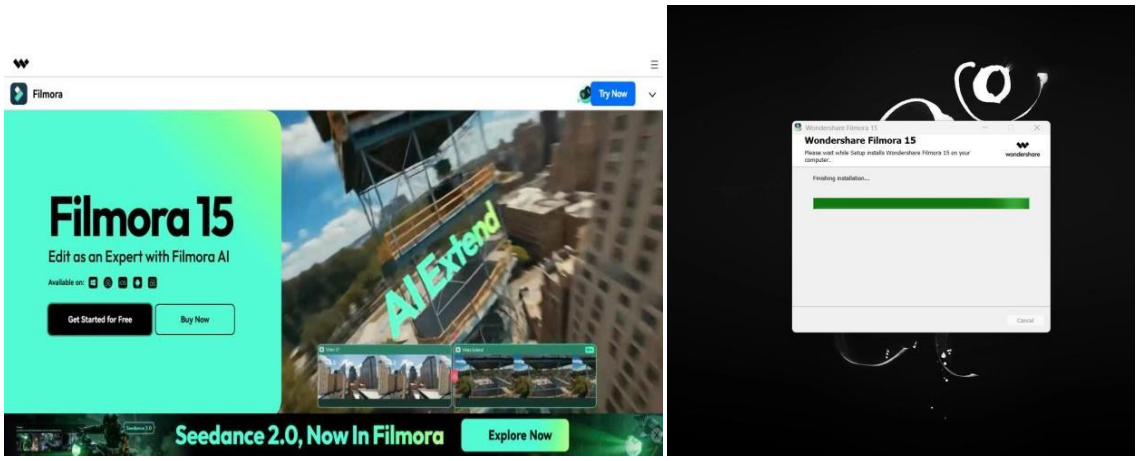

Figure 6. Result – Software Installation

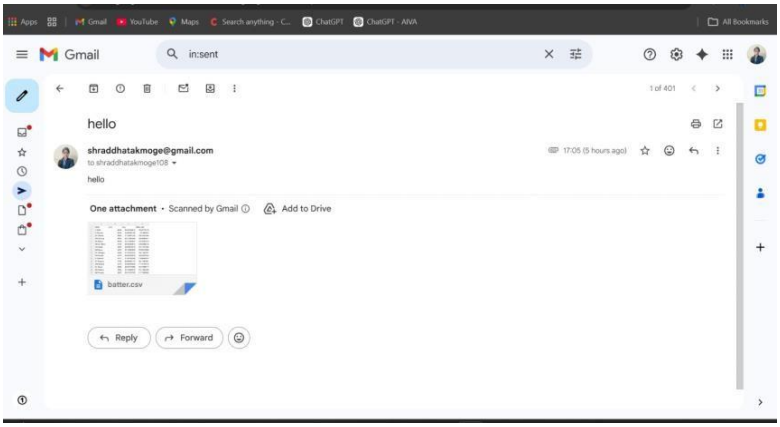

Figure 7. Result – E-mail Automation

X. CONCLUSION

By integrating core AI capabilities such as local speech recognition, natural language understanding, and automated task execution, AIVA provides a highly efficient and portable desktop assistant. The system successfully meets its primary objectives of delivering a seamless voice-first interface, automating routine computer operations, and reducing manual effort for users.

Furthermore, the implementation of dynamic path resolution and localized LLM processing ensures that the system is not only robust and responsive but also easily shareable across different environments without technical friction. The successful replacement of standard terminal outputs with natural vocal feedback creates a complete, hands-free loop that enhances productivity and accessibility. As AI technology continues to evolve, AIVA serves as a scalable foundation for future enhancements in personalized desktop automation and intelligent human-computer interaction.

ACKNOWLEDGMENT

We sincerely thank our guide and HOD, Prof. Vikas Gaikwad , for his expert guidance, continuous support, and for providing the necessary resources at Shree Ramchandra College of Engineering. We also express our gratitude to the faculty of the Department of Artificial Intelligence & Data Science for their encouragement. Finally, we thank our families and friends for their unwavering motivation throughout this project.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 13, April 2026
Impact Factor: 8.2

REFERENCES

[1]. Pankaj Kunekar et al., "AI-Based Desktop Voice Assistant," IEEE ICNTE Conference, 2023.
[2]. Nivedita Singh et al., "Voice Assistant Using Python," International Journal of Innovative Research in Technology, Vol. 8, July 2021.)
[3]. Vishal Kumar Dhanraj et al., "Research Paper on Desktop Voice Assistant," IJRES, 2022.
[4]. Yingfang Zhang et al., IOP Journal of Physics: Conference Series, 2018. [5]. OpenWeatherMap API Documentation – https://openweathermap.org/api [6]. NewsAPI Documentation – https://newsapi.org