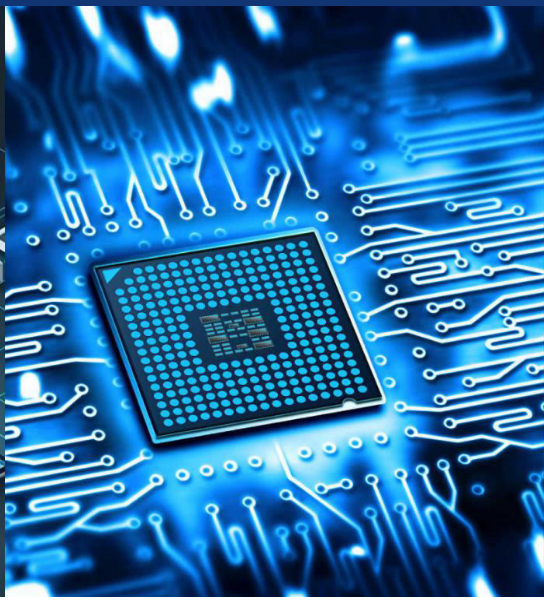
IJIRCCE

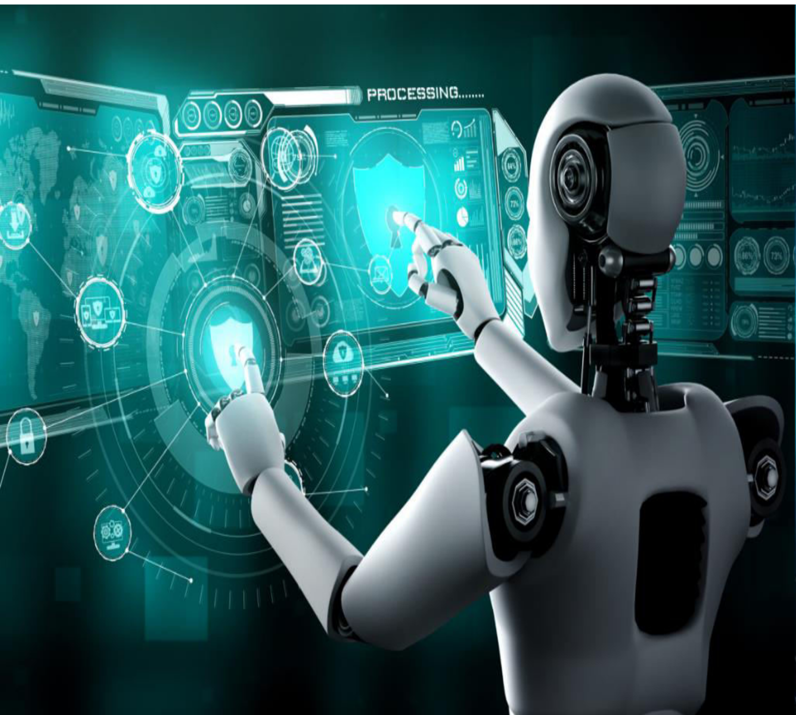
ISSN(O): 2320-9801

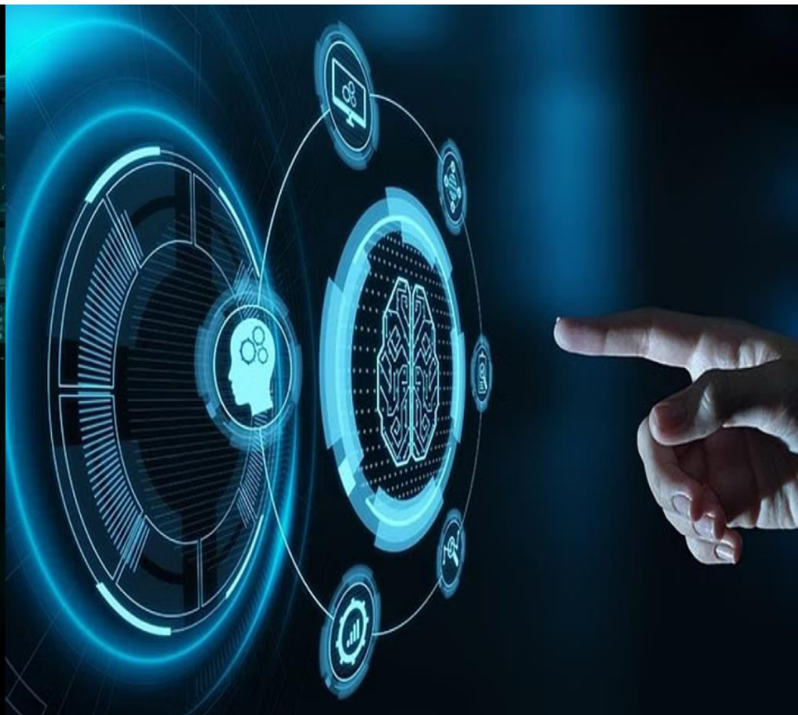
ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)





Impact Factor: 8.771

Volume 14, Issue 4, April 2026

www.ijircce.com ijircce@gmail.com +91-9940572462 +91 63819 07438

© 2026 IJIRCCE | Volume 14, Issue 4, April 2026 DOI: 10.15680/IJIRCCE.2026.1404138

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Smart AI Voice Assistant

Dr.Sayed Akhtar, Shriyash A. Sulakhe, Sanchit R. Umbarkar, Sagar S. Nikam, Akash S. Kale

Professor, Department of CSE, International Centre of Excellence in Engineering & Management, Chh. Sambhaji Nagar, Maharashtra, India

BE Student, Department of CSE, International Centre of Excellence in Engineering & Management, Chh. Sambhaji Nagar, Maharashtra, India

BE Student, Department of CSE, International Centre of Excellence in Engineering & Management, Chh. Sambhaji Nagar, Maharashtra, India

BE Student, Department of CSE, International Centre of Excellence in Engineering & Management, Chh. Sambhaji Nagar, Maharashtra, India

BE Student, Department of CSE, International Centre of Excellence in Engineering & Management, Chh. Sambhaji Nagar, Maharashtra, India

ABSTRACT: This is a Hindi voice-controlled desktop assistant designed and developed for the Windows operating system using Python 3.11 and the PyQt6 graphical framework. The system addresses a significant gap in current voice assistant technology — the lack of native Hindi language support for offline and online PC automation tasks. While global assistants such as Siri, Alexa, and Google Assistant have achieved widespread adoption, they remain primarily English centric and do not cater effectively to the hundreds of millions of Hindi-speaking users in India who prefer to interact with their computers in their mother tongue.

KEYWORDS: Voice Assistant, Hindi NLP, Speech Recognition, Desktop Automation, Python, PyQt6, Devanagari, Hinglish, Natural Language Processing, Wake Word Detection

I. INTRODUCTION

The rapid evolution of Artificial Intelligence and Natural Language Processing has fundamentally transformed human computer interaction. Voice assistants have emerged as one of the most compelling manifestations of this transformation, enabling users to communicate with machines in natural spoken language rather than through conventional keyboard and mouse interfaces. Systems such as Amazon Alexa, Apple Siri, Google Assistant, and Microsoft Cortana have achieved widespread adoption across consumer and enterprise domains, demonstrating the viability and desirability of conversational computing. This a Hindi Voice Controlled Desktop Assistant — is developed to address this gap. The system is a comprehensive, production-grade desktop application built entirely in Python that enables Hindi-speaking users to control their Windows PC entirely through natural Hindi voice commands. The assistant is activated through the wake word "Hey AI" after which it enters a continuous listening mode and accepts commands in Hindi Devanagari script.

II. LITERATURE REVIEW

The academic study of voice-controlled computing dates to the early 1950s with systems capable of recognizing isolated spoken digits. The field has undergone three distinct transformational epochs: rule-based phoneme matching (1950s– 1980s), statistical modelling via Hidden Markov Models (1980s–2010s), and deep learning-based end-to-end neural recognition (2010s–present).

The following survey reviews relevant prior art across commercial assistants, Hindi NLP research, and desktop automation frameworks. IJCRT1601009 International Journal of Creative Research Thoughts (IJCRT) www.ijcrt.org 25 www.ijcrt.org © 20XX IJCRT | Volume X, Issue X Month Year | ISSN: 2320-2882 IJCRT1601009 International

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Journal of Creative Research Thoughts (IJCRT) www.ijcrt.org 26 Amazon Alexa (2014): Alexa introduced the concept of always-listening wake word detection combined with a cloud-based natural language understanding pipeline. Its skill-based extensibility model demonstrated the commercial viability of third-party command domains.

However, Alexa is primarily designed for smart home and e-commerce contexts and does not provide Windows desktop control capabilities. Hindi support remains limited to information queries. The literature review reveals a clear research gap: no existing system provides comprehensive Hindi voice control for Windows desktop computers with both online and offline operation, Devanagari normalization, multi-provider AI integration, and a production-grade GUI. RAM AI addresses this gap through: A novel three-layer Hindi normalization pipeline (phrase-level Devanagari, wordlevel Devanagari, Hinglish romanized) that operates deterministically without neural translation overhead. A microphone conflict resolution architecture using a singleton MicManager that prevents concurrent microphone access by the wake word detector and conversation listener threads

III. PROPOSED METHODOLOGY AND DISCUSSION [Overall Architecture]

This follows a layered, modular architecture in which each layer has a well-defined responsibility and communicates with adjacent layers through defined interfaces. This design facilitates independent development, testing, and future extension of individual components. The architecture comprises seven primary layers arranged in a processing pipeline:

Layer	Description
1. Input Layer	Captures voice input via the microphone (SpeechRecognition / sounddevice) or text input via the GUI keyboard field.
2. Wake Word Layer	WakeWordDetector monitors the audio stream for the wake phrase 'Hey Ram' and triggers conversation mode
3.Recognition Layer	Listen Worker captures post-wake-word speech and submits it to Google Speech Recognition API for transcription.
4.Normalization Layer	hindi_normalizer.py converts Devanagari script, Hinglish phrases, and removes filler words to produce English router keywords.
5. Routing Layer	CommandRouter matches normalized text against regular expression patterns and dispatches to handler functions.
6. Execution Layer	System control handlers (system_control.py, screenshot.py, youtube_module.py, productivity handlers) execute the requested operation
7. Response Layer	SpeechEngine synthesizes the Hindi response text to audio via gTTS or pyttsx3 and emits it through the speaker.

IV. RESULTS

This was tested using a three-tier testing strategy: unit testing of individual module functions, integration testing of the normalization-routing pipeline, and user acceptance testing of the end-to-end voice interaction experience. Given the nature of voice input and TTS output, automated testing was combined with structured manual testing sessions.

Unit Testing: Unit tests were implemented using Python's built-in unittest framework. The following units were individually tested:

- hindi_normalizer.normalize_hindi(): 45 test inputs covering Devanagari phrases, individual Devanagari words, Hinglish romanized phrases, English commands (must remain unchanged), and mixed inputs.
- CommandRouter.route(): 62 test inputs verifying correct handler dispatch and appropriate return types.
- AIBrain._offline(): 15 Hindi, English, and Hinglish conversational inputs verified against expected response patterns.
- SystemControl._fmt(): 10 byte-size inputs verified against expected human-readable strings.
- is_stop_command(): 20 inputs including all defined stop phrases and non-stop commands.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Integration Testing :

TC#	Test Input (Voice / Text)	Expected Normalized	Expected Action	Result	Status
TC-01	वॉÐयम बढTओ	volume up	Volume increases	As expected	PASS
TC-02	वॉÐयम घटTओ	volume down	Volume decreases	As expected	PASS
TC-03	आवTज बंद करो	mute volume	Volume muted	As expected	PASS
TC-04	यूट्यूब खोलो	open youtube	YouTube opens	As expected	PASS
TC-05	øि◌ो◌म खोलो	open chrome	Chrome opens	As expected	PASS
TC-06	Öøiनशॉट लो	take screenshot	Screenshot saved	As expected	PASS
TC-07	ि◌◌ोसÖटम जTनकTरी	system info	CPU/RAM/Disk shown	As expected	PASS
TC-08	मौि◌◌ो◌ोम बतTओ	weather	Weather opens	As expected	PASS
TC-09	ि◌◌ो◌ोमय बतTओ	time	Current time spoken	As expected	PASS

V. CONCLUSION

This is Hindi Voice Controlled Desktop Assistant demonstrates that it is technically feasible to build a production-grade, Hindi first voice assistant for the Windows desktop using entirely open-source and freely available Python libraries. The project makes three primary technical contributions: (1) a novel three-layer Hindi normalization pipeline that converts Devanagari script and Hinglish romanized speech recognition output into English command keywords without neural translation overhead; (2) a microphone conflict resolution architecture using a singleton MicManager that enables simultaneous background wake word monitoring and foreground conversation listening without hardware conflicts; and (3) a cascaded TTS system that provides natural Hindi audio output across a range of software availability scenarios from the ideal (gTTS online) to the universal (Windows SAPI, available on all Windows installations without additional packages).

REFERENCES

[1] Zhang, Y., et al. (2022). "A Survey on Voice Assistant Technology: History, Architecture, and Future." ACM Computing Surveys, 54(4), 1-42.
[2] Choudhary, N., et al. (2018). "Improving Hindi Automatic Speech Recognition Using Transfer Learning." Proceedings of Interspeech 2018, Hyderabad, India.
[3] Dua, M., et al. (2019). "Improving Hindi Automatic Speech Recognition System using Transliteration." Journal of King Saud University – Computer and Information Sciences, 31
(4), 505-512. [4] Patel, R., et al. (2021). "Hinglish Language Processing: Challenges and Approaches." Proceedings of the 2021 Conference on Language, Data and Knowledge, Vienna, Austria.
[5] Kumar, P., et al. (2020). "Design and Development of a Hindi Voice-Activated Assistant." International Journal of Computer Applications, 175(32), 28-34.
[6] Google LLC. (2023). "Cloud Speech-to-Text API Documentation." Retrieved from https://cloud.google.com/speechto-text/docs

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE
SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

Scan to save the contact details

www.ijircce.com