International Journal of Computational Research in Science and Technology
E-ISSN: 3108-0871, Volume 2 Issue5
Sep-Oct- 2026, PP 01-07

AI-Powered Voice Assistants: A Review Of Speech Recognition, Language Understanding, Speech Synthesis And Security

Khebade Anushka Santosh[1]; Prof. M. C. Kolse[2]; Prof. S. K. Said[3]
Department Of Artificial Intelligence And Data Science Jaihind College Of Engineering Pune, Maharashtra, India;

Abstract—An AI-powered voice assistant lets people interact with computers and devices through natural speech rather than keyboard and mouse. It recognizes spoken commands, understands the user's intent, performs an action and replies with synthesized speech. This paper reviews the end-to-end speech pipeline that makes voice assistants possible. It covers wake-word detection with small-footprint neural networks, automatic speech recognition from early deep acoustic models to recurrent, transformer-based and self-supervised systems trained on large and diverse audio, spoken language understanding through intent classification and slot filling, dialogue and action execution, and neural text-to-speech that produces natural voices. Evaluation measures such as word error rate and intent accuracy are explained, and the security of voice interfaces against hidden and inaudible commands is examined. A five-layer architecture comprising audio capture, speech recognition, language understanding, action execution and speech output is proposed, along with a research gap analysis, objectives and a comparison of speech recognition approaches. Accessibility benefits for students, professionals and people with disabilities are discussed. The review concludes that modern speech and language models enable simple, hands-free and efficient human–computer interaction, provided that accuracy for diverse accents, privacy and voice security are addressed.

Index Terms—Voice assistant, speech recognition, speech-to-text, text-to-speech, natural language processing, wake-word detection, human–computer interaction, voice security

I. Introduction

1.1 Motivation

Speaking is the most natural way for people to communicate, and voice assistants bring this mode of interaction to computers, phones, cars and homes. A user can ask for the time, search the internet, open an application or play music without touching a screen. Hoy [1] describes how assistants such as Siri and Alexa made voice interaction mainstream, and López *et al.* [2] show that their usefulness depends on how well they understand varied requests.

Building a voice assistant requires solving several problems in sequence: detecting when the user is speaking to the device, converting speech to text, understanding the meaning, performing the requested action and responding with natural speech. Each of these stages has been transformed by deep learning over the last decade.

This paper reviews the technologies behind AI-powered voice assistants, proposes a modular architecture, identifies gaps and discusses challenges. It complements text-based assistant studies by focusing on the speech pipeline and its specific difficulties.

1.2 Objectives of the Review

The objectives of this review are:

• To study each stage of the voice assistant pipeline and its key techniques.

International Journal of Computational Research in
Science and Technology
E-ISSN: 3108-0871, Volume 2 Issue5
Sep-Oct- 2026, PP 01-07

• To compare traditional, end-to-end and self-supervised speech recognition approaches.

• To propose a modular voice assistant architecture.

• To examine evaluation metrics and security threats specific to voice interfaces.

• To identify challenges and future directions, including multilingual and offline operation.

II. Overview of Key Concepts

2.1 The Speech Pipeline

Fig. 1 shows the components of a voice assistant. A wake-word detector listens continuously for a trigger phrase; automatic speech recognition (ASR) converts the following speech to text; spoken language understanding (SLU) determines intent and parameters; an action module performs the task; and text-to-speech (TTS) produces the spoken reply.

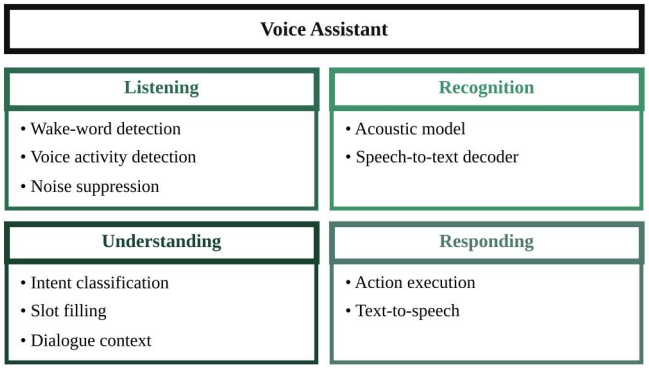


Fig. 1. Components of an AI-powered voice assistant.

2.2 Speech Representation

Audio is sampled, divided into short frames and converted into features such as Mel-frequency spectrograms. Neural networks map these features to characters, sub-words or words. Standard corpora such as LibriSpeech [3] are used to train and compare ASR systems.

III. Review of Existing Work

Wake-word detection. Chen *et al.* [4] proposed small-footprint keyword spotting with deep neural networks that run with low memory and latency on devices, outperforming earlier hidden Markov model approaches. Warden [5] released the Speech Commands dataset to standardize research on limited-vocabulary recognition.

Speech recognition. Hinton *et al.* [6] reported that deep neural networks substantially reduced error rates compared with Gaussian mixture models, marking the deep learning era in speech. Graves *et al.* [7] showed that deep recurrent networks trained end-to-end achieved strong results on phoneme recognition. The Conformer [8] combined convolution and self-attention to capture both local and global audio patterns. Self-supervised wav2vec 2.0 [9] learned representations from unlabelled audio and achieved good accuracy with very little labelled data, valuable for low-resource Indian languages.

Whisper [10], trained on a very large, multilingual and weakly labelled corpus, generalizes robustly across accents, background noise and languages.

Language understanding. Tur and De Mori [11] formalized SLU as intent detection and slot filling, and Chen *et al.* [12] performed both jointly with BERT.

Speech synthesis. WaveNet [13] generated raw audio waveforms with a deep autoregressive network, producing much more natural speech than concatenative systems, and Tacotron 2 [14] combined a sequence-to-sequence spectrogram predictor with a WaveNet vocoder to approach human quality.

Security. Carlini *et al.* [15] crafted hidden voice commands that sound like noise to humans but are recognized by assistants, and Zhang *et al.* [16] demonstrated DolphinAttack, using ultrasonic signals inaudible to people to issue commands. Table I summarizes the reviewed work.

Sr. No.	Ref.	Year	Methodology	Findings	Research Gap
1	[6]	2012	Deep neural network acoustic models	Large error reduction over GMM-HMM	Still needs HMM pipeline
2	[7]	2013	Deep bidirectional LSTM for speech	Strong end-to-end phoneme recognition	Not streaming-friendly
3	[4]	2014	Small-footprint DNN keyword spotting	Low-latency on-device wake-word detection	Fixed keyword set
4	[13]	2016	WaveNet autoregressive audio model	Highly natural synthesized speech	Slow original sampling
5	[16]	2017	Inaudible ultrasonic commands	Exposed microphone non-linearity attack	Requires nearby attacker
6	[14]	2018	Tacotron 2 + WaveNet vocoder	Near-human TTS quality	Needs high-quality voice data
7	[8]	2020	Conformer (convolution + attention)	State-of-the-art ASR accuracy	Compute intensive
8	[9]	2020	wav2vec 2.0 self-supervised learning	Good accuracy with little labelled data	Large pre-training cost
9	[10]	2023	Whisper large-scale weak supervision	Robust multilingual recognition	Large models; can hallucinate text

TABLE I Summary of Literature Survey

IV. Identified Research Gaps

The following gaps are identified:

• Recognition accuracy drops for regional accents, code-mixed speech and noisy environments common in India.

• Many assistants depend on cloud processing, raising privacy concerns and failing without internet.

• Voice interfaces remain vulnerable to hidden, inaudible and replayed commands.

International Journal of Computational Research in Science and Technology
E-ISSN: 3108-0871, Volume 2 Issue5
Sep-Oct- 2026, PP 01-07

• Context across multiple turns is handled poorly by simple command-based assistants.

• Evaluations seldom include users with speech impairments or older adults.

V. System Architecture

The proposed five-layer architecture is shown in Fig. 2. Wake-word detection and basic commands run on the device, while complex recognition and queries can use cloud services when available.

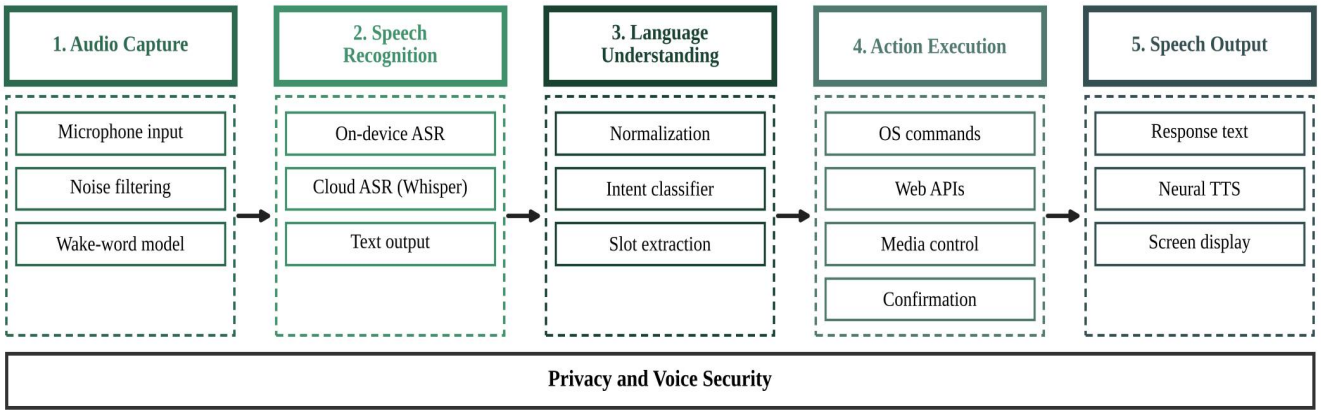


Fig. 2. Proposed five-layer architecture of the AI-powered voice assistant.

5.1 Audio Capture Layer

A microphone captures audio, which is filtered for noise and checked by voice activity detection. A small on-device network listens for the wake word, and only audio after the wake word is processed further.

5.2 Speech Recognition Layer

A speech-to-text engine converts the command into text. An on-device model handles common commands offline, while a larger model such as Whisper can be used for open-ended queries.

5.3 Language Understanding Layer

The text is normalized and classified into an intent, such as get_time, search_web, open_app or play_music, and slots such as the application name or search query are extracted.

5.4 Action Execution Layer

Handlers perform the requested task by calling operating-system functions, web APIs or media players. Sensitive commands, such as sending messages or making payments, require confirmation or speaker verification.

5.5 Speech Output Layer

The response text is converted to natural speech by a neural TTS engine and played to the user, with a text display for accessibility.

International Journal of Computational Research in Science and Technology
E-ISSN: 3108-0871, Volume 2 Issue5
Sep-Oct- 2026, PP 01-07

VI. Implementation Methodology

The working of the assistant is shown in Fig. 3. ASR accuracy is measured by the word error rate

$$WER = (S + D + I) / N \quad (1)$$

where S, D and I are the numbers of substituted, deleted and inserted words, and N is the number of words in the reference transcript. ASR models are commonly trained with the connectionist temporal classification objective, which sums over all alignments π of the audio x to the transcript y:

$$P(y \mid x) = \Sigma_{\pi} \Pi_t P(\pi_t \mid x) \quad (2)$$

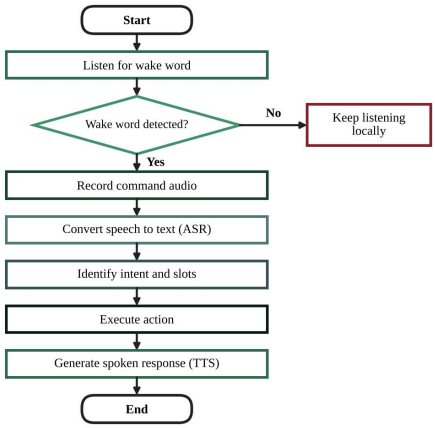


Fig. 3. Working of the voice assistant.

End-to-end response time, from end of speech to start of the spoken reply, is measured because delays above about one second make interaction feel unnatural.

VII. Performance Comparison

Table II compares speech recognition approaches. Classical HMM-based systems need expert-designed components; end-to-end neural models simplify the pipeline; self-supervised models need little labelled data; and large weakly supervised models are the most robust but require more computation.

TABLE II Comparison of Speech Recognition Approaches

Approach	Accuracy	Data Need	Deployment
GMM-HMM	Low–Medium	Medium	Light
DNN-HMM hybrid	Medium–High	High	Moderate
End-to-end RNN / CTC	High	High	Moderate
Conformer / Transformer	Very high	High	Heavy
Self-supervised (wav2vec 2.0)	High	Low labelled	Heavy
Weakly supervised (Whisper)	Very high, robust	Very high	Heavy; small variants exist

VIII. Open Issues and Future Directions

Table III summarizes challenges and mitigations. Future voice assistants will run capable speech and language models on-device, support many Indian languages and code-mixing, recognize emotion and context, and integrate with smart-home and IoT devices. Their accessibility benefits for people with visual or motor impairments [17] make inclusive design an important research direction.

TABLE III Challenges and Mitigation

Challenge	Mitigation
Accents and noise	Diverse training data; noise suppression
Internet dependence	On-device ASR for common commands
Privacy of recordings	Local wake word; process minimal audio
Voice attacks	Speaker verification; ultrasonic filtering
Multilingual users	Multilingual and code-mixed models

IX. Conclusion

AI-powered voice assistants make human–computer interaction more natural and accessible. The reviewed literature shows how deep learning improved each stage of the speech pipeline: compact networks detect wake words, end-to-end and self-supervised models recognize speech robustly, transformers understand intent and neural vocoders generate natural speech. A modular architecture that combines on-device processing for speed and privacy with cloud models for complex requests, while defending against voice attacks, provides an effective, user-friendly assistant for students, professionals and anyone who prefers hands-free interaction.

9.1 Acknowledgment

The authors express their sincere gratitude to the Department of Artificial Intelligence and Data Science, Jaihind College of Engineering, Pune, for providing the facilities, guidance and encouragement needed to carry out this work. The authors also thank the faculty members and colleagues whose valuable suggestions helped improve the quality of this review.

References

1. M. B. Hoy, “Alexa, Siri, Cortana, and more: An introduction to voice assistants,” Med. Ref. Serv. Q., vol. 37, no. 1, pp. 81–88, 2018.
2. G. López, L. Quesada, and L. A. Guerrero, “Alexa vs. Siri vs. Cortana vs. Google Assistant: A comparison of speech-based natural user interfaces,” in Proc. AHFE, Springer, 2017, pp. 241–250.
3. V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “LibriSpeech: An ASR corpus based on public domain audio books,” in Proc. IEEE ICASSP, 2015, pp. 5206–5210.
4. G. Chen, C. Parada, and G. Heigold, “Small-footprint keyword spotting using deep neural networks,” in Proc. IEEE ICASSP, 2014, pp. 4087–4091.
5. P. Warden, “Speech commands: A dataset for limited-vocabulary speech recognition,” arXiv:1804.03209, 2018.

6. G. Hinton et al., “Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups,” IEEE Signal Process. Mag., vol. 29, no. 6, pp. 82–97, 2012.
7. A. Graves, A. Mohamed, and G. Hinton, “Speech recognition with deep recurrent neural networks,” in Proc. IEEE ICASSP, 2013, pp. 6645–6649.
8. A. Gulati et al., “Conformer: Convolution-augmented transformer for speech recognition,” in Proc. Interspeech, 2020, pp. 5036–5040.
9. A. Baevski, H. Zhou, A. Mohamed, and M. Auli, “wav2vec 2.0: A framework for self-supervised learning of speech representations,” in Proc. NeurIPS, vol. 33, 2020, pp. 12449–12460.
10. A. Radford et al., “Robust speech recognition via large-scale weak supervision,” in Proc. ICML, 2023, pp. 28492–28518.
11. G. Tur and R. De Mori, Spoken Language Understanding: Systems for Extracting Semantic Information from Speech. Chichester, U.K.: Wiley, 2011.
12. Q. Chen, Z. Zhuo, and W. Wang, “BERT for joint intent classification and slot filling,” arXiv:1902.10909, 2019.
13. A. van den Oord et al., “WaveNet: A generative model for raw audio,” arXiv:1609.03499, 2016.
14. J. Shen et al., “Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions,” in Proc. IEEE ICASSP, 2018, pp. 4779–4783.
15. N. Carlini et al., “Hidden voice commands,” in Proc. 25th USENIX Security Symp., 2016, pp. 513–530.
16. G. Zhang et al., “DolphinAttack: Inaudible voice commands,” in Proc. ACM CCS, 2017, pp. 103–117.
17. A. Pradhan, K. Mehta, and L. Findlater, ““Accessibility came by accident”: Use of voice-controlled intelligent personal assistants by people with disabilities,” in Proc. CHI, 2018, Paper 459.