

Automated Personal Voice Based Assistant by Using Python

Author: CHALLA BHUPATHI RAJU(MCA student), M. BALA NAGA BHUSHANAMU² (Asst. Prof)

Department of Information Technology & Computer application,

Andhra University College of Engineering, Visakhapatnam, AP.

Corresponding Author: Challa Bhupathi Raju

(email-id:challa.bhupati@gmail.com)

ABSTRACT: In today's digital age, voice assistants have become an integral component of human-computer interaction, enabling users to interact with smartphones, tablets, and smart devices in a seamless, hands-free manner. These AI-powered assistants—such as Google Assistant, Amazon Alexa, and Apple Siri—have revolutionized the way individuals manage their daily routines by offering functionalities like sending messages, checking the weather, setting reminders, controlling IoT devices, and retrieving online information through natural voice commands. However, one of the significant limitations of these mainstream assistants is their reliance on continuous internet connectivity and cloud-based artificial intelligence processing. As a result, their usability significantly diminishes in environments with limited or no internet access, making them less suitable for rural, remote, or offline use cases.

KEYWORDS: Artificial Intelligence, Speech Recognition, Text-To-Speech, Java, XML, Android Studio, Natural Language Processing (NLP) Techniques: Tokenisation, Normalization, Pattern Matching, Keyword Matching, Rule based Decision Making.

I. INTRODUCTION

This toy was truly simple, wherein a toy pooch would stay inside a canine house until the stoner blurted its name, “ Rex ” at which point it would jump out of the house. This was each done by an electromagnet tuned to the frequency analogous to the vowel set up in the word Rex, and anteceded ultramodern computers by over 20 times. In the 21st century, mortal commerce is being replaced by robotization truly snappily. One of the main reasons for this change is performance. In moment's world, we train our machines to do their tasks by themselves or to suppose like humans using technologies like Machine knowledge, Neural Networks, etc. Now in the current period, we can talk to our machines with the help of virtual sidekicks. Virtual assistants are software programs that help you ease your day to day tasks, such as showing weather reports, giving daily news, searching the internet etc. computational approach of this project revolves around the seamless integration of voice recognition, natural language processing, command interpretation, and IoT device control. The assistant starts by continuously listening for user input via a microphone, capturing audio data. This audio input is processed using either Google Speech Recognition API in online mode or the Vosk speech model in offline mode. These tools convert speech into text using advanced speech-to-text (STT)

algorithms. Once the spoken command is converted into a string, the assistant parses the text using simple rule-based logic or keyword matching to determine the user's intent. This parsing step involves identifying action keywords such as "turn on", "open", "play", or "shut down", and mapping them to predefined functions. After understanding the command, the assistant categorizes it into one of several action types — system-level operations (like opening applications, shutting down the system), informational queries (such as telling time, fetching jokes or facts), utility services (like setting reminders or playing music), or IoT device control through Arduino For system commands, Python's os and subprocess libraries are used to execute tasks on the computer. For controlling Arduino-connected devices, the assistant sends specific instructions through a serial communication channel using the pyserial library.

II. Related work:

This is substantially because of its demand in bias like smartwatches or fitness bands, speakers, Bluetooth earphones, mobile phones, laptop or desktop, TV, etc. nearly all the digital bias which are coming currently are coming with voice sidekicks which help to control the device with speech recognition only. A new set of ways is being developed constantly to ameliorate the performance of voice automated hunt. As the quantum of data is adding

exponentially now known as Big Data the stylish way to ameliorate the results of virtual sidekicks is to incorporate our sidekicks with machine literacy and train our bias according to their uses. Other major ways that are inversely important are Artificial Intelligence, Internet of effects, Big Data access and operation, etc. With the use of voice sidekicks, we can automate the task fluently, just give the input to the machine in the speech form and all the tasks will be done by it from converting your speech into textbook form to taking out keywords from that textbook and execute the query to give results to the stoner. Machine literacy is just a subset of Artificial Intelligence. This has been one of the most helpful advancements in technology. Before AI we were the bones who were upgrading technology to do a task but now the machine is itself suitable to fight new tasks and break it without need to involve the humans to evolve it. This has been helpful in day-to-day life. From mobile phones to particular desktops to mechanical diligence these sidekicks are in veritably important demand for automating tasks and adding effectiveness.

III. System Analysis and Design

3.1 Introduction

The aim of this project is to develop a voice assistant system that enables users to interact with computers and IoT devices using natural spoken language. The system should function both online (via Google Speech Recognition) and offline (via Vosk), offering flexibility, privacy, and usability in areas with limited internet access. It must also support Arduino-based device control via serial communication.

Functional Requirements:

1. Voice Recognition: Capture user speech and convert it into text.
2. Wake Word Detection: Optionally activate the assistant with a specific trigger word.
3. Command Parsing: Identify the user's intent based on recognized speech.
4. System Execution: Perform actions like opening apps, shutting down, or searching the web.
5. IoT Control: Communicate with Arduino to control devices like LEDs, motors, or sensors.
6. Feedback Delivery: Respond to the user through text-to-speech output.

7. Mode Switching: Toggle between online and offline voice recognition.

8. Continuous Listening: Maintain a loop for multi-command interactions.

9. Error Handling: Handle unrecognized or invalid commands gracefully.

✗ Non-Functional Requirements:

- Performance: Fast response time and low latency.
- Scalability: Easy to add new commands or integrate more devices.
- Usability: Simple interface and voice-friendly behavior.
- Portability: Should run on any desktop OS (Windows/Linux).
- Security: Offline mode ensures data privacy.

3.2 Software and Hardware Requirements

3.2.1 Software Requirements:

- ✓ Python 3.5 & Above
- ✓ Windows 7 And Above

3.2.2 Hardware Requirements:

- ✓ Processor: Intel Core i5
- ✓ RAM: 4GB
- ✓ OS: Windows / Mac
- ✓ Microphone
- ✓ ARDUINO UNO board
- ✓ Relay
- ✓ A Light Bulb
- ✓ USB Cable
- ✓ Electronics Wires
- ✓ Plug Point & a Plug

3.3 Design Approaches

- a) The voice adjunct takes an input word which is called as "signal word" to be actuated. so, it takes in the signal word and starts operating for the stoner commands.
- b) Converting the speech into textbook will be reused by the adjunct.
- c) The converted textbook is now reused to get the needed results.
- d) The textbook given by the stoner should contain one or two keywords that determine what query is to be executed. If the keyword doesn't match any of the queries in the law also the adjunct asks the stoner to speak again. e) Eventually, the affair to the stoner's query will be given by converting speech to textbook. VOICE Adjunct Our adjunct "NOVA" extends to helps us when working on a system in which it's installed. We can pierce by calling the wake word "Hello

NOVA". WHAT IS VOICE ASSISTANT A voice adjunct, also known as an intelligent particular adjunct or a connected speaker, is a new type of device that's grounded on natural language speech recognition and is offered by popular companies like Apple, Amazon, and Google. We got spired by that and created one our tone. WHY DO WE NEED IT generally, codifying out and searching or doing day- to- day tasks becomes excited. But our life doesn't need to be like that. One can ask for help to state sidekicks. They let the druggies to perform a task using a speech command, as well as recoup information via voice conflation. Following are the reasons to have a voice adjunct.

- minimum trouble It's easier to say a many words than class them on a small smartphone screen.
- Eyes Free One can be as eyeless as a club, but a voice adjunct will always help you. Our cognizance are enough. One can also ask the bot about commodity while cooking at the same time.
- Fast response Imagine how important time you have to spend to find some information on a website? Or how numerous clicks do you need to make before you find the thing you need in a mobile operation? Voice sidekicks don't induce similar difficulties. One can ask a question and you have the answer.

WHERE TO USE IT :

Voice hunt has been a hot content of discussion. Voice visibility will really be a challenge. This is due to the lack of a visual interface for voice sidekicks. druggies can not see or interact with a voice interface unless it's linked to the Alexa or Google Assistant app. Search geste patterns will change dramatically as a result. Brands are presently witnessing a metamorphosis in which touch points are transubstantiating into harkening points, and organic hunt will be the primary means of brand visibility. Advertising agencies are getting more popular as voice hunt grows in fashionability. Voice sidekicks will also continue to offer further personalized gests as they get better at secerning between voices. The number of people using voice sidekicks is anticipated to grow. According to the Voice bot Smart Speaker Consumer Adoption Report 2018, nearly ten percent of people who don't enjoy a smart speaker plan to buy one. However, the stoner base of smart speaker druggies will grow 50 percent, meaning a quarter of grown-ups in the United States will enjoy a smart speaker, If this holds true. integrating real-time deep learning, sensor fusion, and control systems.

IV. PROPOSED METHODOLOGY

4.1 Objective of the Project

Main ideal of erecting particular adjunct software(a virtual adjunct) is using semantic data sources available on the web, stoner generated content and furnishing knowledge from knowledge databases. The main purpose of an intelligent virtual adjunct is to answer questions that druggies may have. This may be done in a business terrain, for illustration, on the business website, with a converse interface. On the mobile platform, the intelligent virtual adjunct is available as a call- button operated service where a voice asks the stoner " What can I do for you? " and also responds to verbal input.

Virtual sidekicks can extensively save you time. We spend hours in online exploration and also making the report in our terms of understanding. give a content for exploration and continue with your tasks while the adjunct does the exploration. Another delicate task is to flash back test dates, birthdates or anniversaries. It comes with a surprise when you enter the class and realize it's class test moment. Just tell adjunct in advance about your tests and she reminds you well in advance so you can prepare for the test. One of the main advantages of voice quests is their velocity. In fact, voice is reputed to be four times faster than a written hunt whereas we can write about 40 words per nanosecond, we're able of speaking around 150 during the same period of time. In this respect, the capability of particular sidekicks to directly fete spoken words is a prerequisite for them to be espoused by consumers.

4.2 Methodology

The methodology of the design follows a structured channel that enables voice- grounded commerce with a computer system and IoT bias(via Arduino). The adjunct processes stoner voice input, converts it into textbook, analyzes the intent, and executes the corresponding action — either system- position commands or tackle control. This approach uses a modular design, allowing inflexibility and easy integration of unborn advancements like new commands, detectors, or APIs. sphere the internet of effects, or IoT, is a system of interrelated computing bias, mechanical and digital machines, objects, creatures or people that are handed with unique identifiers(UIDs) and the capability to transfer data over a network without taking mortal- to- mortal or mortal- to- computer commerce. A thing in the internet of effects can be a person with a heart examiner implant, a ranch beast with a biochip transponder, an machine that has builtin detectors to warn the motorist when tire pressure is low or any other natural or man- made object that can be assigned an Internet protocol(IP) address and is suitable to transfer data over a network. Decreasingly, associations in a variety of diligence are using IoT to operate more efficiently, more understand guests to deliver enhanced client service, ameliorate decision- making and increase the value of the business. An IoT ecosystem consists of web- enabled smart bias that use bedded systems, similar as processors, detectors and communication tackle, to collect, shoot and act on data they acquire from their surroundings. IoT bias partake the detector data they collect by connecting to an IoT gateway or other edge device where data is either transferred to the pall to be analysed or analysed locally. occasionally, these bias communicate with other affiliated bias and act on the information they get from one another. The bias do utmost of the work without mortal intervention, although people can interact with the bias-- for case, to set them up, give them instructions or access the data.

4.3 Programming Languages

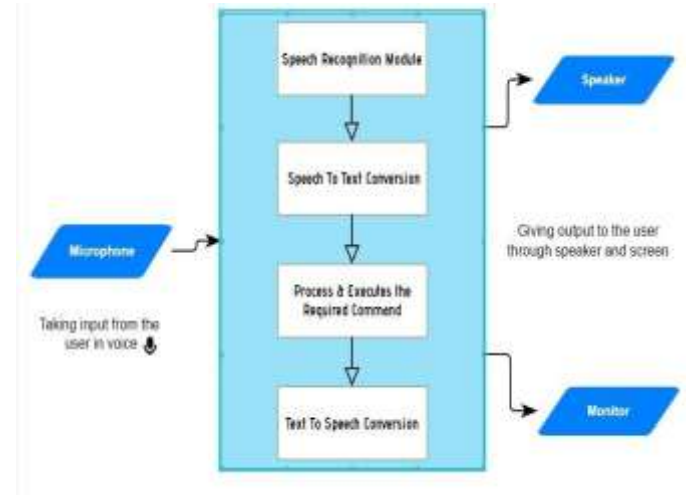
Python is an OOPs(Object acquainted Programming) grounded, high position, interpreted programming language. It's a robust, largely useful language concentrated on rapid-

fire operation development(RAD). Python helps in easy jotting and prosecution of canons. Python can apply the same sense with as important as 1/ 5th law as compared to other OOPs languages. The operation of Python is similar that it can not be limited to only one exertion. Its growing fashionability has allowed it to enter into some of the most popular and complex processes like Artificial Intelligence(AI), Machine literacy(ML), natural language processing, data wisdom etc. Python has a lot of libraries for every need of this design. For this design, libraries used are speech recognition to fete voice, Pyttsx for textbook to speech, selenium for web robotization etc. Easy to be told and perceive- The syntax of Python is simpler; thence it's comparatively straightforward, indeed for newcomers jointly, to be told and perceive the language. Multi-purpose language – Python could be a multi-purpose programing language as a result of it supports structured programming, object- acquainted programming yet as practical programming. Support of open force community – As being open force programing language, Python is supported by awfully giant inventor community. Because of this, the bugs square measure simply mounted by the Python community.

4.4 Algorithm Used

Specch Recognition Module The class which we're using is called Recognizer. It converts the audio lines into textbook and module is used to give the affair in speech. Energy threshold function represents the energy position threshold for sounds. Values below this threshold are adjusts the energy threshold stoutly using audio from source(an audio source case) to regard for ambient noise Speech To Text & Text To Speech Conversion Pyttsx3 is a textbook- to-speech conversion library in Python. And can change the Voice, Rate and Volume by specific commands. Python provides an API called Speech Recognition to allow us to convert audio into textbook for farther processing converting large or long audio lines into textbook using the Speech Recognition API in python. We've Included sapi5 and espeak TTS Machines which can reuse the same. Process & Executes The needed Command The said command is converted into textbook via speech recognition module and farther stored in a temp. also, dissect the stoner's textbook via temp and decide what the stoner needs grounded on input handed and runs the while circle. also, Commands are executed.5.2 Pseudocode

4.5 Flow Chart



VI. IMPLEMENTATION

The successful implementation of a voice assistant system integrated with IoT depends on the use of a combination of software and hardware technologies. This project leverages widely available and open-source tools, making it suitable for educational, personal, and customizable use cases. The core technologies used include Python, Arduino, speech recognition, text-to-speech, and serial communication.

6.1 Python Programming Language

Python is the main programming language used to develop the voice assistant due to its simplicity, flexibility, and extensive libraries. Python provides modules for handling audio input/output, processing commands, interacting with hardware, and executing system tasks. It supports both offline and online capabilities, making it an ideal language for voice-based automation projects.

6.2 Speech Recognition (Google API & Vosk)

The system uses two modes of speech recognition:

Google Speech Recognition API: An online service that converts spoken language into text with high accuracy. Requires internet access and is used when a reliable connection is available.

Vosk (Offline STT Engine): An open-source speech recognition toolkit that allows for completely offline speech processing. It enables the assistant to work in environments without internet access and helps maintain privacy.

These tools are integrated using the speech_recognition library in Python.

6.3 Text-to-Speech (pyttsx3)

To provide vocal responses, the assistant uses pyttsx3, a text-to-speech library in Python. This allows the system to speak out responses to the user's commands. It works offline and supports voice customization, which helps personalize the assistant's output (e.g., saying the user's name).

6.4 PySerial for Arduino Communication

The assistant communicates with Arduino using the pyserial library, which enables serial communication over USB. Commands such as "L_ON" or "F_OFF" are sent from the Python program to the Arduino board to control LEDs,

motors, and other connected devices. This forms the bridge between software and physical hardware (IoT).

6.5. Arduino UNO (IoT Controller)

Arduino is used as the hardware interface to control physical devices like LEDs, motors, sensors, and other electronic components. It reads incoming serial commands and triggers specific actions using digital I/O pins. The Arduino sketch is written in C/C++ and is uploaded via the Arduino IDE.

6.6. System Libraries and Utilities

Additional libraries such as os, datetime, and subprocess are used for:

- Opening files and applications
- Fetching system time/date
- Performing background operations like launching web browsers or media players this enables the assistant to perform system-level tasks beyond IoT control.

6.4 Web Page with Drive Video Selected

This image displays the web interface in which a user has chosen the video file drive.mp4 for upload. The system is running locally using Flask and is accessible at 127.0.0.1:5000. The page features a simple and user-friendly design with a central upload panel labeled "Upload a Driving Video." It is built to make the interaction straightforward, enabling users to submit road footage with just one click. Once selected, the video is transferred to the backend server, where lane detection processing begins.

VIII. REFERENCES

Easwara Moorthy, Aarthi & Vu, Kim-Phuong, "Voice Activated Personal Assistant: Acceptability of Use in the Public Space" HIMI 2014. Lecture Notes in Computer Science, vol **8522**. Springer, pp. 324-334, 10.1007/9783-319-07863-2_32.

V. Kēpuska and G. Bohouta, "Next-generation of virtual personal assistants

(Microsoft Cortana, Apple Siri, Amazon Alexa and Google Home)," 2018 *IEEE 8th Annual Computing and Communication Workshop and*

Conference (CCWC), 2018, pp. 99-103, doi: 10.1109/CCWC.2018.8301638.

George Terzopoulos and Maya Satratzemi. 2019. "Voice Assistants and Artificial Intelligence in Education". In Proceedings of the 9th Balkan Conference on Informatics (BCI'19). Association for Computing Machinery, New York, NY, USA, Article 34, 1–6.

DOI:<https://doi.org/10.1145/3351556.3351588>

R. Kumar, G. Sarupria, V. Panwala, S. Shah and N. Shah, "Power Efficient Smart Home with Voice Assistant," 2020 11th *International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 2020, pp. 1-5, doi: 10.1109/ICCCNT49239.2020.9225612.

T. -K. Kim, "Short Research on Voice Control System Based on Artificial Intelligence Assistant," 2020

International Conference on Electronics, Information, and Communication (ICEIC), 2020

pp. 1-2, doi: 10.1109/ICEIC49074.2020.9051160.

S. Subhash, P. N. Srivatsa, S. Siddesh, A. Ullas and B. Santhosh, "Artificial Intelligence-based Voice Assistant," 2020 *Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*, 2020, pp. 593-596, doi: 10.1109/WorldS450073.2020.9210344.

Dellaert, B.G.C., Shu, S.B., Arentze, T.A. et al. Consumer decisions with artificially intelligent voice assistants. *Mark Lett* 31, 335–347 (2020).

<https://doi.org/10.1007/s11002-020-09537-5>

Harshit Agrawal, Nivedita Singh, Gaurav Kumar, Dr. Diwakar Yagyasen, Mr. Surya Vikram Singh "Voice Assistant Using Python" 2021, *IJIRT* Volume 8 Issue 2, ISSN: 2349-6002, pp.419-423.

Abeed Sayyed, Ashpak Shaikh, Ashish Sancheti, Swikar Sangamner, Prof. Jayant H Bhangale. "Desktop Assistant AI Using Python" (2021) *International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)* Volume 6, Issue 2, June 2021. ISSN (Online) 2581-942.

P. Krishnaraj, F. Mohamed Faris, D. Rajesh "Portable Voice Recognition with GUI Automation" 2021, *IJIRT*, Volume 9 Issue 6 ISSN (Print): 2320-9356 PP. 20-23

Mrs. A.M. Sermakani, J. Monisha, G. Shrisha, G. Sumisha, "Creating Desktop Speech Recognition Using Python Programming." *IJARCCCE*, Vol. 10, Issue 3, March 2021, ISSN (Online), pp.129-134.

Rajdip Paul, Nirmalya Mukhopadhyay "A Novel Python-based Voice Assistance System for reducing the Hardware Dependency of Modern Age Physical Servers". *IRJET*, Volume: 08 Issue: 05, May 2021, e-ISSN: 2395-0056, ISSN: 2395-0072.

V Geetha & Gomathy, C K & Kottamasu, Manasa & Kumar, Nukala. (2021). The Voice Enabled Personal Assistant for Pc using Python. *International Journal of Engineering and Advanced Technology*. 10. 162-165. 10.35940/ijeat.D2425.0410421.

Zwakman, Dilawar Shah & Pal, Debajyoti & Triyason, Tuul & Arpnikanondt, Chonlameth. (2021). Voice Usability Scale: Measuring the User Experience with Voice Assistants. 308-311. 10.1109/iSES50453.2020.00074. Sprengholz, Philipp & Betsch, Cornelia "Ok Google: Using virtual assistants for data collection in psychological and behavioral research" (2021) *Behavior Research Methods*. DOI:10.3758/s13428-021-01629-y

Buhalis D., Moldavska I. "In-room Voice-Based AI Digital Assistants

Transforming On-Site Hotel Services and Guests' Experiences" 2021 In: Wörndl W., Koo C., Stienmetz J.L. (eds) *Information and Communication Technologies in Tourism 2021*. Springer, Cham.

https://doi.org/10.1007/978-3-030-65785-7_3, pp.