AIJR Preprints
Section: Working Paper
Article Id: 423, Version: 1, 2022
DOI: https://doi.org/10.21467/preprints.423
URL: https://preprints.aijr.org/index.php/ap/preprint/view/423

{Click on above link or DOI to see the latest available version of this article}

NOT PEER-REVIEWED

Version 1: Received: 23 November 2022 / Approved: 04 December 2022 / Online: 06 December 2022

Python Based User Interactive Voice Bot

Akshay Khare

Department of Electronics and Telecommunication, Vishwakarma Institute of Technology, Pune, India, 411038

Corresponding Author Email Id: akshaykhare0501@gmail.com

ABSTRACT

In this paper, a python-based voice bot is proposed which is able to give answers to any question of the user and it should be able to follow commands of the user. AI is the main area of creating intelligent machines, particularly computer programs. It is related to the well-known task of employing computers to comprehend human intelligence. This paper provides an outline of the concept of a personal assistant for Windows-based platforms. The system is inspired by virtual assistants such as Siri for iOS and Cortana for Windows. It has been intended for performing a variety of operations by utilizing well-defined commands. This work focuses on assisting end-users with day-to-day activities such as searching questions on Google, general conversation with the user, searching for YouTube or other websites, automatic keyboard and mouse handling, sending mail, mathematical calculations, opening applications and files on the PC, clicking photos, and many more tasks. Machine learning is used to assess user statements/commands to provide the best solution.

Keywords: Python, Speech recognition, Sqlite3 database

1 Introduction

Nowadays, the development of artificially intelligent systems capable of generating machine-human Interaction through communication, voice, actions, body language, and so on. It is getting importance. The way of interaction, based on the knowledge of the machine by the machine of natural language processing (NLP), is among the most seen and famous.[1] It is no longer the case that a human needs self-learning to interact with a machine, but rather that a machine does self-learning to speak with a person, presenting its movements, routines, and behavior while attempting to become his greatest personal assistant. Work on creating and improving these personal assistants has been ongoing for a long time. Such systems are continuously evolving and moved across personal computers, and already made a strong presence in a variety of mobile gadgets and devices.[2]

Voice assistants such as Google Assistant, Cortana, Alexa, and Siri can help you with tasks such as starting applications, looking for information, sending messages, and more by converting users' commands i.e., audio into text. The process of turning audio to text is known as speech recognition [3].

How to Cite:

Akshay Khare, "Python Based User Interactive Voice Bot". *AIJR Preprints*, 423, Version 1, 2022.

Python is a popular programming language for developers and programmers. The proposed voice bot is created using python. Python has a Speech-recognition API that allows us to turn audio to text for additional processing.[4] The queries for assistance can be customized to meet the needs of the user. This voice bot functions similarly to Google Assistant or Cortana. This voice bot can speak with the user or interact with the user by replying to questions of the user or performing activities which the user wants him to do. However, it is not very advanced, it is entirely dependent on the if-else conditions that have been used in the program.

This paper provides an overview of the design and building of a voice bot. It includes a work plan as well as the technique for a voice bot's work. It also explains the testing outcomes. The fundamental purpose of this effort is to create a local voice bot that can perform human tasks and routine work which humans must perform on a daily basis.

2 Literature Review

Every company developer of an intelligent assistant has his/her own design process, which influences the final product. Some assistants can provide more descriptive speech, some assistants can execute duties more efficiently and without additional descriptions and modifications, while some assistants can do a smaller set of tasks more efficiently and as per the user's requirements. But there is no universal assistant capable of performing all duties equally well. The range of functionality that an assistant has is entirely dependent upon which domain the developer has given the most focus to. As all systems are machine learning-based methods that employ large amounts of data acquired from many resources and then trained on them, the provider of this data, whether it is search systems, other information resources, or social media networks, plays an important role.[1] The form of the helper is determined by the amount of data gathered from various resources. Although there are many methods of learning, algorithms, and procedures, the phenomenon of developing these systems is nearly identical. Text to speech, natural language comprehension, automatic speech recognition, voice biometrics, named entity recognition, and Voice activation are the major technologies.

Ashutosh Sakharkar et al. proposed a python-based AI assistant for computers. In their work, machine learning is used to analyze user commands. Their voice assistant can perform general human interactions, search queries on Google, search for medicines information, retrieve videos and photos, vocabulary help, remind the user about upcoming tasks and events. Also, it is able to tell live weather forecast details as well as health suggestions based on symptoms.[1]

Selvan et al. proposed a python based interactive voice bot. In their system, when the gadget is turned on, the user speaks into the microphone, and the input is sent. It requires vocal input from the user and then sends it to the computer, which then leads towards the next process. Instead, it sends the speech to the text converter, which converts it into textual form as the output, which is finally located and recognized. The text is then analyzed, as well as the parameters are checked. The voice recognition system is built around the procedure of the parameter, which tests keyword matches for the text. If the keywords match, it returns the correct output. It is then transformed to speech output by using a text to speech converter that employs character recognition technology. OCR categorizes and identifies the text, which is then converted into audio by the text to the voice engine. The output is sent via the mics connected to the raspberry pi's headphone connector.[5]

An intelligent voice recognition chatbot is proposed by S. J. du Preez et al. The web-based bot produces personalized user responses that are associated with the desired character. Queries asked to the bot that are not grasped are analyzed further by an online intelligent research assistant, and the response is preserved, strengthening the artificial brain's skills for future response creation.[6]

Divakar et al. suggested a solution which is a smartphone application created to help farmers in two ways, a speech bot and a recommendation bot. They constructed an agricultural voice bot using pysttsx3, Google translator, and Google search engines to react to farmer requests in multiple languages. They also have a recommendation bot that may provide a variety of suggestions in response to a farmer's queries about the climate, crops, soil, fertilizers, and so on. Farmers can use this smartphone application to improve farming techniques and maximize agricultural yield.[7]

Google Assistant, Apple Siri, Amazon Alexa, Microsoft Cortana, and Samsung Bixby are some popular examples of voice assistants. These voice assistants allow the devices to fade into the background, effectively making technology "invisible." In everyday use, voice assistants fulfill this duty by responding to general queries. A study in 2018 by SEO and digital marketing firm Stone Temple (now Perficient Digital) compared the completeness and accuracy of answers to over 4,000 queries to measure the intellect of personal assistants. Google Assistant, Amazon Alexa, and Microsoft Cortana take the lead, with Siri coming in the fourth position.[8]

3 Methodology

A. Requirements:

All the required libraries and modules have been installed using the command prompt. The code of the proposed system has been written in Spyder which is an integrated development environment (IDE) for scientific programming in the Python language [9]. DB Browser is required for creating a database and storing data in it [10]. For executing some tasks, a web browser is required.

B. Implementation Details:

After successful installation of all required libraries and modules, all these libraries and modules have been imported into the code. The work begins with an analysis of the voice commands delivered by the user through the microphone. It can include anything from searching for information to operating contents in a computer. To include more functions in the voice bot, many internet sources and documentation of various python libraries have been used. Fig. 1 depicts the detailed process of the voice bot's basic workflow. For speech recognition purposes, the Speech Recognition library in python is used. It provides simple audio processing as well as access to the microphone. It converts speech to text, which the machine understands. To convert output text to speech, the pyttsx3 module of python is used [11]. In the proposed system, sapi5 speech API is used which is developed by Microsoft. It helps in the synthesis and recognition of voice [12]. The pyttsx3 module supports two voices, one first is a male voice and another is a female voice, which is provided by sapi5 for Windows operating system. Voice id helps us to select different voices. For male voice, “voices [0].id” is used and for female voice “voices [1].id” is used. Then the text is passed to the central processor, which recognizes the command and invokes the appropriate script for execution.

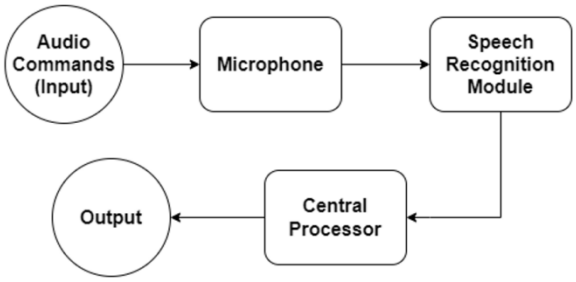

Fig.1.: *Basic Workflow*

However, the issues do not stop here. Aspect like background noise can play a significant impact in the process of speech recognition. Voice recognition lacks the ability to distinguish the ambient noises it listens to from your speech, such as the sound of song or television playing in the background. To filter out those ambient noises, a noise removal method is used. Another aspect is how people naturally vary the frequency of their speech to compensate for noisy surroundings; speech recognition algorithms can be sensitive to such pitch shifts in most cases. In the proposed system, the input process continues until the end. Though the user has finished his speech input, the recognizer waits for an adequate amount of time before switching and converting the input into the text. The Speech Recognition module then finishes the recognizing process and generates a searching thread. The processes in the proposed model are shown in Fig. 2.

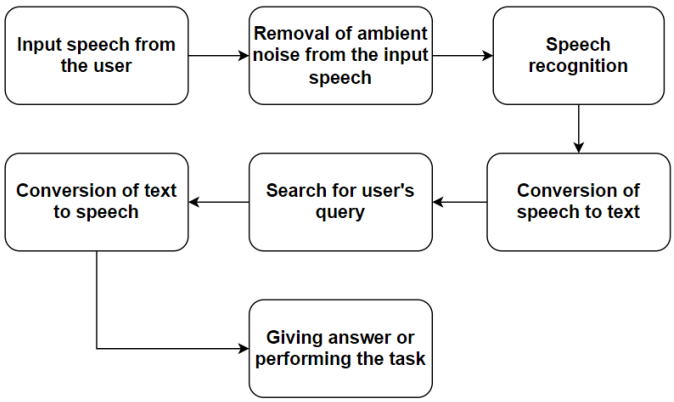

Fig.2.: *Processes in the model*

C. Working

In the proposed system, each input query is checked in 'if-else' conditions. As shown in Algorithm 1, in each condition, some keywords have been added. One by one each 'if-else' condition will be checked for the user's query. If these keywords are found in the user's command or query, then the voice assistant will check these keywords in each 'if-else' condition till the requirement is properly satisfied. Then it will give an answer or perform the task accordingly.

Suppose the user asks a question "What is today's date". Another user asks, "Tell me today's date". In both of these questions, 'today' and 'date' are common words. Also, the user can ask the same question in a different manner, but these two words will be common in each of those questions. So, in spite of including all of those questions for one answer, in the 'if' condition, only 'today' and 'date' can be included.

Algorithm 1: Working of voice bot

```
Input: query by the user
Output: answer or task performed by the voice bot
Initialization:
said = true
while(said)
    if keywords in query then
        return answer or perform task
    else if 'wh' or 'how' in query then
        return answer page on google
    else if 'stop' in query then
        said = false
        end while
    else
        return answer from Wikipedia
    insert query in the database
end
```

If the user asks any question to the voice bot and if its answer is already given in the code, then it will speak that one or do the respective task which user wants. If this condition does not satisfy, then it will be checked that if the user's query is a rhetorical question or not. If it is a rhetorical question i.e., it contains 'wh' or 'how' keywords, then the voice bot will search that question on google and open the answer page on google. If the question asked by the user does not satisfy the above conditions, then finally that query will be searched through Wikipedia. For this purpose, the Wikipedia library is installed and imported into the code. If the user's command contains the keywords like 'stop', 'bye', or 'quit' then the voice bot will come to know that the user wants to quit the voice bot. So, the voice bot will stop, and it will pop up a window that asks for a review. Then the user has to select an option according to his/her satisfaction with the performance of the voice bot. This review will be stored in a local database which is created using SQLite3. Also, after answering each query of the user, each query is recorded in the local database.

To handle errors effectively, 'try-except' conditions have been used.

D. Features

The voice bot is able to do conversations with the user. It is able to do nineteen different kinds of calculations. These operations are addition, subtraction, multiplication, division, modular division, calculating square, square-root, cube, cube-root, factorial, percentage, simple interest, power or exponent of a number, and six kinds of trigonometric operations i.e., sin, cos, tan, cot, cosine and sec. This voice assistant has two voices. i.e., male voice and female voice. The user can switch those voices at any time. The voice bot is able to give the answer to any question. It can open any website, YouTube, Google. The user has to tell the name of that website to the voice assistant to open that website. It is able to open any app and file on the user's device. e.g., Word, Excel, WhatsApp, Image Gallery, etc. It is able to play music. It is able to tell us the current time and current, tomorrow, yesterday and some other dates. It is able to tell the name of the current month and year. Also, it can show the calendar for any month in any year. For this purpose, datetime and calendar modules have been used. If the user gives a command to wait for a specific period of time, then the voice bot will wait for the user for that

time period. It is able to send mail to anyone using the smtplib module which uses Simple Mail Transfer Protocol (SMTP) [13]. It can click photos. For this purpose, OpenCV has been used [14]. The user can set an alarm and after the set time reaches, the voice bot rings the alarm. The user can play the Tic-Tac-Toe game as well as some online games. For creating alarm and Tic-Tac-Toe game, the Tkinter library of python is used which helps in making Graphical User Interface (GUI) [15]. The voice bot can perform brightness control, sound control, minimizing or closing the window, taking screenshots, etc. In addition to this, the voice bot can perform some automatic keyboard and mouse handling functions using the pyautogui module of python [16].

4 Results

```
Listening...
Recognizing...
User said: who is the Prime Minister of India

Listening...
Recognizing...
User said: close this window

Listening...
Recognizing...
User said: yes

Listening...
Recognizing...
User said: click photo

Listening...
Recognizing...
User said: click photo

opencv_frame_0.png written!
Listening...
Recognizing...
User said: close the camera
```

Fig.3: *Proper working of the voice bot*

In Fig. 3, the execution of the voice bot is shown. While testing this voice bot, most of the answers and performed tasks by the voice bot are proven to be correct as per the user's requirement.

5 Conclusion

Through this voice bot, various services have been automated by just using voice commands. It simplifies most of the user's duties, such as searching the web, taking images, performing mathematical computations, and so on. In the proposed system, the voice bot is developed for only English to English conversation. The multilingual voice bot can be developed using python programming. Also, using Django, Flask, or some other framework, this voice bot can

be converted into a website. In the future, speech recognition will change the way people transact business on the internet in the future, eventually integrating world-class e-business.

References

[1] Sakharkar, Ashutosh, Shidesh Tondawalkar, Pratik Thombare, and Rajashri Sonawane. "Python Based AI Assistant for Computer." (2021).
[2] Shende, Deepak, Ria Umahiya, Monika Raghorte, Aishwarya Bhisikar, and Anup Bhange. "AI Based Voice Assistant Using Python." Journal of Emerging Technologies and Innovative Research 6, no. 2 (2019): 506-509.
[3] AbouElhasan, Nourhan, Tasneem Elboraee, Hamees Mohamed, Nourhan Adel, and Marwa M. Eid. "SURVEY ON AUTOMATIC SPEECH RECOGNITION." Journal of Computer Science and Information Systems 11, no. 2 June 2020 (2020).
[4] Apturkar, Akash, Alexander I. Iliev, Amruth Anand, Arush Oli, Swadesh Reddy Siddenki, and Vikram Reddy Meka. "Sentiment Analysis of Speech with Application to Various Languages."
[5] Selvan, Mercy Paul, M. S. Roobini, S. Jancy, Suji Helen, Viji Amutha Mary, and Angajala Preethi Servani. "Interactive Bot Using Python." NVEO-NATURAL VOLATILES & ESSENTIAL OILS Journal| NVEO (2021): 1501-1507
[6] du Preez, Salomon Jakobus, Manoj Lall, and Saurabh Sinha. "An intelligent web-based voice chat bot." In IEEE EUROCON 2009, pp. 386-391. IEEE, 200
[7] Divakar, M. S., Vimal Kumar, Martina Jaincy DE, R. A. Kalpana, and Sanjai Kumar RM. "FARMER'S ASSISTANT using AI Voice Bot." In 2021 3rd International Conference on Signal Processing and Communication (ICPSC), pp. 527-531. IEEE, 2021.
[8] Reviews.com Staff. (2020). The Best Voice Assistants
[9] Hussain, Zahid, and Muhammad Siyab Khan. "Introducing Python Programming for Engineering Scholars." International Journal of Computer Science and Network Security 18, no. 12 (2018): 26-33.
[10] Hillar, Gaston C. Building RESTful Python Web Services. Packt Publishing Ltd, 2016.
[11] Sripriya, N., S. Poornima, S. Mohanavalli, R. Pooja Bhaiya, and V. Nikita. "Speech-Based Virtual Travel Assistant for Visually Impaired." In 2020 4th International Conference on Computer, Communication and Signal Processing (ICCCSP), pp. 1-7. IEEE, 2020.
[12] Yadav, Avanish Vijaybahadur, Sanket Saheb Verma, and Deepak Dinesh Singh. "VIRTUAL ASSISTANT FOR BLIND PEOPLE." INTERNATIONAL JOURNAL 6, no. 5 (2021).
[13] Goerzen, John. "Simple Message Transport Protocol." In Foundations of Python Network Programming, pp. 197-210. Apress, Berkeley, CA, 2004.
[14] Mordvintsev, Alexander, and K. Abid. "Opencv-python tutorials documentation." 2014.
[15] Beniz, Douglas, and Alexey Espindola. "Using Tkinter of python to create graphical user interface (GUI) for scripts in LNLS." WEPOPRPO25 9 (2016): 25-28.
[16] Sweigart, Al. "PyAutoGUI Documentation." Read the Docs (2020): 25.