[[1]](#footnote-1)

JARVO: Voice Assistant

ARNAV SHARMA E19CSE398; PRANJAL SINGH E19CSE401; TUSHAR MARWAH E19CSE399; VATSAL AGARWAL E19CSE289

# INTRODUCTION

Voice assistant is a program that responds to verbal instructions given by the user. A Voice Assistant is a system assistant that listens to specific voice commands and returns some of important information or performs specificed actions as required by the user by using speech recognition, language processing and voice synthesis. These programs are on digital devices that instructs a program to process human speech into a text format. Voice assistant is a technology that allows us a hands-free control over a machine and that too in a different variety of languages. Advanced voice technology will soon be commonplace allowing us to have natural conversations on our phones. We will have natural spoken conversations with our smart phones in the not-too-distant future. Virtual Assistants, often known as Intelligent Personal Assistants, include Voice Assistants. These virtual assistants may accept input in a variety of formats. A virtual assistant that employs voice recognition, natural language processing, and speech synthesis to aid its users is known as a Voice Assistant.

MOTIVATION

There has been a lot of difficulty for users who are unable to type on a machine maybe due to some physical disability. Another reason might be lack of knowledge or simply if a person is illiterate and is unable to write text on a machine in a proper format without errors. Considering small kids who are just new learners who just hears words used all around from beginning from their parents. Time is also a major concern. A lot of time is consumed to type text, emails, Wikipedia searches and much more on a machine which becomes quite boring and irritating at the same time.

SOLUTION

To solve all these problems, we have got our new A.I. assistant-J.A.R.V.O. It is a user-friendly voice assistant which is surely a bridge between a man and a machine. Voice assistant quickly types the words much faster than a human as soon as they are spoken. It will be useful for people who have some disability and are unable to type fast. Kids who are in their learning stage will be able to use this voice assistant for much easier ways of learning as the kids pick up a pattern, syntax, tone and starts to discover more. Also, it would be so easier for the user to send mails without typing anything, searching on Wikipedia, Google, YouTube, or any other search engine without even opening it. Playing music without typing the name of the song will be much easier. User can also open code editor or IDE. This all can be done with just a single voice command. This would make life easier and will save a lot of time.

## II. RELATED WORK

The ability to talk is the most important feature of an artificial intelligence assistant. We'll create a function called speak to make our voice assistant talk. This function takes an audio file as an input and then pronounces it.

Example:

def speak(audio):

The audio is the next item we'll need. We need to provide audio so that we may use the speak () method to pronounce it. We're going to install the pyttsx3 module.

What exactly is pyttsx3?

A Python module to aid in the conversion of text to speech. It is, in a nutshell, basically this is text to speech library. It is compatible with Python and works offline.

We created a function named speak (). To transform our text to speech, we'll now construct the speak () method.

Making Our Main () Function: First, we'll make a main() function, and then we'll call our talk function from within it.

#example

if \_\_name\_\_=="\_\_main\_\_" :

speak("hello I am jarvo")

Everything you type into the speak () method will be turned to speech. Congratulations! Our assistant now has its own voice and is prepared to speak.

Wishme() Function:

Now we'll define the Wishme() function, which will make our assistant greet the user based on the computer time. We need to load a module named datetime to supply current time to A.I.

They should look for voice data collecting services that can be accessible fast and easily via an API, such as:

• The Google Cloud Speech-to-Text API

• Nuance's Automatic Speech Recognition (ASR) technology

Following that, you design and create software to meet your needs. Python may be used to create algorithms and modules.

OTHER ALGORITHM

**Speech Recognition in the Car**

It's not only about making things simpler with speech-activated devices and digital voice assistants. It's also about security – at least in terms of in-car voice recognition.

• You can now instruct your car who to contact or which restaurant to go to instead of texting while driving.

If your car's gasoline tank is going low, you’re in-car speech system may not only tell you that you need to fill up, but also guide you to the nearest gas station and ask if you have a favourite brand. Or it would ask you that the petrol station you selected is very far away to reach with amount of the fuel you have left.

III BLOCK DIAGRAM

Chart, diagram

Description automatically generated

PROPOSED METHODOLOGY

**Features**

* To find anything on Wikipedia, type the following into the search box:

Installing and importing the Wikipedia module into our software is required to do Wikipedia searches.

Import the Wikipedia module into the app using an import statement when it has installed.

* To open the YouTube website in a web browser:

To start any webpage, we must first import the web browser module. It is a built-in module, so we don't need to use the pip command to install it; instead, we can use the import command to import it right into our application.

We're using an elif loop here to see if Youtube is in the user's search. Assume that the user issues the command "Jarvo., open youtube." As a result, elif confition will become true and it will open youtube.

* To open the Google website in a web browser:

We're using the same reasoning to open Google in a web browser as we did to load YouTube.

* To play music

To play music, we'll need to import the os module. Use an import statement to directly import this module.

With the aid of the os module, the music directory was opened and all of the songs in it were listed. You may play any music you want with the aid of os.startfile. J.A.R.V.O will play the song randomly from the list.

* to find out what time it is

The datetime() method is used to start the code to save the current or live system time in a variable named strTime. This variable is given an argument. The time string will now be transformed into speech.

* To open the VS Code Program

elif 'open code' in query:

The code route of the program is necessary. It is important for opening VS Code or any other application.

The following are the steps to obtain the application's code path:

Step 1: Go to the file's location and open it.

Step 2: Right-click the application and select properties from the menu.

Step 3: Take the target from the section and paste it into the target section.

Save the application's target into a variable once you've copied it. I'm storing the target into a variable named codePath, and then we're opening the programme with the os module.

* **Sending mails**

To mail anyone, smtplib module is imported. The Simple Mail Transfer Protocol is a protocol for sending and routing emails between mail servers. An instance method named sendmail is present. We may send an email with this instance method. There are three factors to consider:

• The sender: This is the sender's email address.

• The recipient: The recipient's email address.

• The message: A multi-recipient string message.

Defining the Send Email feature

We'll construct a sendEmail() method to assist us in sending emails to one or more recipients.

We're utilising the SMTP module in the preceding code, which we've already examined.

Remember to activate the 'less secure applications' setting in your Gmail account as well. The sendEmail function will not operate correctly if this is not done.

Within the main() function, call the sendEmail() method:

elif 'email to a person' in query:

To address any conceivable issue while sending emails, we use the try and except block.

**Result & Analysis**

At first we used gTTS API(Google text to speech). It is an api which is very easy to use, which converts text into speech.

Features of gTTS-

* Customizable speech-specific sentence tokenizer that can read any length of text while maintaining proper intonation, abbreviations, decimals, and other features;
* Text pre-processors that can be customised to give features such as pronunciation adjustments;
* Languages that are supported are automatically retrieved.

Google Text-to-speech did not give good results as there were problems in importing the API and calling it. Also, it was difficult to build a perfect system.

To overcome these problems we used Pyttsx3 and Sapi5.

pyttsx3 is a library in python which converts text to speech. One of the main advantage of this library is that it does not need internet.

Sapi 5 is an API developed by Microsoft it aids in voice synthesis and recognition.

Result:

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

1. First and foremost, we constructed the wishme() function, which provides greeting functionality based on the time of our A.I. system.
2. We implemented a takeCommand() function after the wishme() function to assist our A.I in taking commands from the user. This method is also in charge of returning a string representation of the user's inquiry.
3. We created the coding logic for several websites such as Google, YouTube, and Stack Overflow.
4. Finally, we added the ability to send emails.

**Comparison with Google Assistant:**

Google Assistant is a virtual assistant developed by Google it employs artificial intelligence.

Unlike Google Now, the company's previous virtual helper, the Google Assistant can hold two-way conversations. Whereas Jarvo is a simple voice assistant that has limited capabilities like sending emails on your behalf, doing wikipedia searches without opening web browser and performing various other tasks like playing music. Also Google Assistant is written in C++ and Jarvo in Python.

**CONCLUSION**

Our voice assistant named jarvo uses specific voice commands and returns some of important information or performs specificed actions as required by the user by using speech recognition, language processing and voice synthesis.

This can be done with just a single voice command. This would make life easier and will save a lot of time.

1. [↑](#footnote-ref-1)