

Voice Recognition(AI): Voice Assistant Robot

Naman Pundir (Author)

Computer Science Department, HMRITM, GGSIPU, DELHI

Abstract — Voice interface is becoming a common feature in devices such as tablets, smartphones and laptops. Moreover, voice recognition technology is touted to mature and become the default method to control of a variety of interfaces, including mobile devices, windows and microcontrollers such as Arduino and raspberry pi. Thus, it is critical to understand the factors that influence the use of voice activated applications in the public domain. The present study examined how the perceived acceptability of using the Voice-Activated Personal Assistant (VAPA) in devices influences its reported use. If the devices can listen to the user for the request or handle the daily affairs, then give the right response, it will be much easier for users to communicate with their devices and the devices will be much more “Smart” as a human assistant. This project is focusing on the Raspberry voice assistant control development over the voice control (recognition, generate and analyze corresponding commands, intelligent responses automatically) that are needed in daily life. Physically disable person can also use this Application. Voice SMS is an application developed in this work that allows a user to record and convert spoken messages into SMS text message. Speech recognition for Voice uses a technique based on hidden Markov models (HMM - Hidden Markov Model). It is currently the most successful and most flexible approach to speech recognition. Therefore, we want to talk about one simple kind of assistant robot that we build it, because this kind of robot give us more help in our daily needs. This paper talks about new method to make robot that assist human in conversational way. After hard working we conclude in this paper the procedure to make assistant robot using Raspberry-Pi three model B, speaker, microphone and supplementary elements.

This project can listen the command and reply by linking with different cloud sources.

Keywords— *HMM (Hidden Markov Model), TTS (Text to Speech), Intent, Raspberry pi, google api.*

I. INTRODUCTION

Many years ago, software programs were developed and run on the computer. Nowadays, smart phones and all other smart devices are widely used by all people. This shows that the market is increasing fast and there are also more capabilities for Smartphone because of this wide use. This project is originated from a popular application from Apple called “Siri” and Google voice Action. Siri application was released on the date when the iPhone4S was published. Google Voice Action can be used in any android version

above 4.4. In this work we have developed an application for sending SMS messages, opening inbuilt Applications like gallery, settings, camera, messaging etc. which uses Google's speech recognition engine. The main goal of application is to use user voice commands to access smart phone instead of using it manually. As it integrates most of the mobile phone services for daily use, it could be useful for getting a more convenient life and it will be helpful for those people who have disabilities for manual operations. This is also part of the reason why it has been chosen as the degree project. In this application user is able to access the services of smart devices with their voice command. User can easily access to all over the internet and can even perform numerous services provided by the devices. In this Project physically disable person or the person having less knowledge about smart devices or how to access them smartly with their voice or speech command.

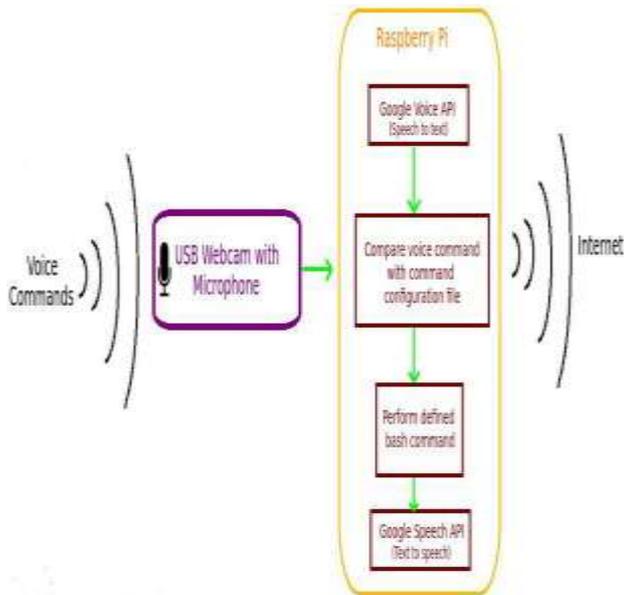
II. NEED OF THE SYSTEM

Back in the 1980"s and 1990"s, home computers didn't rely on dedicated monitors to display the operating systems, applications, and games. Instead of paying extra for a monitor, most console owners were happy to use their televisions only. But as far as computers go, the PC model of dedicated monitor was eventually widely adopted. The problem for a middle-class human to buy the computer system should be taken into consideration. So, the need of a cheap, alternative system raised; where the costly computer system should have an alternative. Also for security purpose of home, small offices, there was a requirement of the system which can provide features such as facial recognition and voice recognition which should be available in very cheap rate and small in size. The rate and the size problem were covered where invented Raspberry Pi model by embedded manufacturers giving extremely super quality of minicomputer. Here arises the need of the system. The Raspberry pi is a series of single board computer developed in wales by raspberry pi foundation. All raspberry pi includes the same video-core IV GPU and either a single core ARMv6 or newer ARMv7 quad-core.it is of 256 or 512 mb RAM. It basically uses Linux kernel base operating systems. The install manager of Raspberry pi is NOOBS. For storage it uses SDHC and micro SDHC.

III. WORKING PROCESS

This software is based on using the Google voice and speech API's. The voice is recorded using the Microphone. Then using Google speech API's, it is converted to text. The text is compared with previously configured commands in configuration command file. The text matches with any of the commands, thus it performs the specific task. After finding the required commands the bash command for task

is executed. The speed of this process is really very fast and depend on the speed of the internet connected.



IV. CHECKING MICROPHONE

Initial step of this process is to check whether the microphone/webcam works properly. We must check whether it records properly and checking microphone high volumes, etc are high. First check whether the USB microphone is connected properly using the command "lsusb". Check whether our mic/webcam comes under the list.

Then checking the recording in high volumes. to do this, type the command "alsamixer" in the terminal. Thus an interface appear, arrow buttons are used to increase the volumes. Select the particular USB mic/webcam from the list and check at high volumes. To check recording is done correctly use the command "arecord -l". use the command "arecord -D plughw:1, 0test.wav" to record sound. The sound is recorded in test.wav.

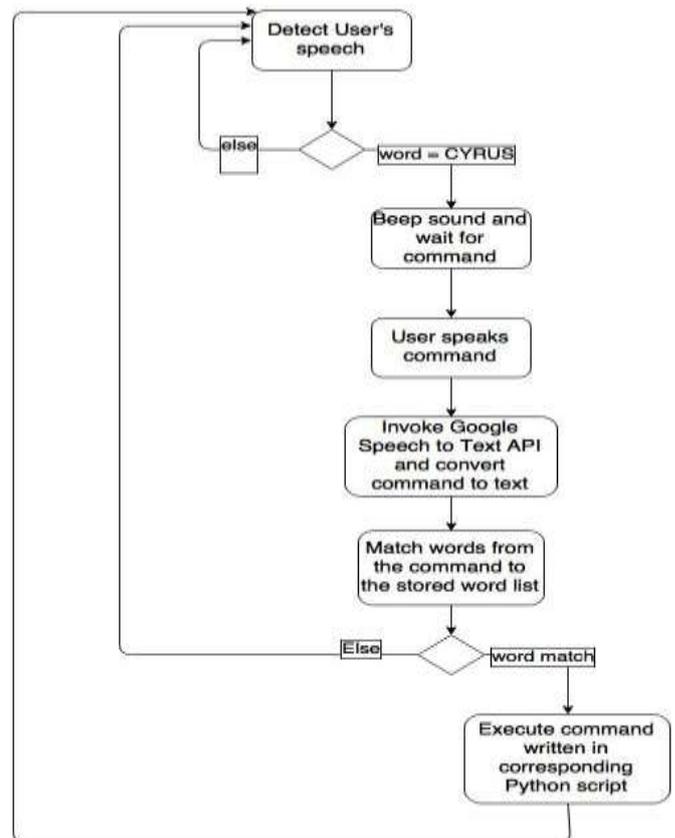
To listen to the recorded sound, plug in your headphones in your raspberry pi device and enter the command "aplay test.wav" in the terminal. Speech Recognition for intelligent voice assistant application is done using Google Server, using HMM algorithm. This process involves the conversion of acoustic speech into a set of words and is performed by software component. Accuracy of speech recognition systems differ in vocabulary size and confusability, speaker dependence vs. independence, modality of speech (isolated, discontinuous, or continuous speech, read or spontaneous speech), task and language constraints. The system consists of five modules: feature extraction, HMM phone model training, dictionary preparation, grammar estimation, and sentence decoding.

Speech recognition systems which are based on Hidden Markov Models which are today's most widely applied in modern technologies. They use the word or phoneme as a unit for modeling. The model output is hidden probabilistic functions of state and can't be deterministically specified. State sequence through model is not exactly known.

V. INSTALLING VOICE RECOGNITION SOFTWARE IN RASPBERRY PI

This software utilises Google Voice API's. To install the software enter the following commands.

- 1) `Wget -no-check-certificate http://goo.gl/KrwrBa -O PiAUISuite.tar.gz`
- 2) `Tar -xvzf PiAUISuite.tar.gz`
- 3) `Cd PiAUISuite/Install/`
- 4) `Sudo ./InstallAUISuite.sh`
- 5) Wget command in first line uses two dashes (--)
- 6) Before no check .During installation several questions may popup. press(y/n) accordingly .some of the questions include do you want to set keyword?.there ill be also questions like do you want to set filler flag to zero?, do you want to install youtube-dl?(to use youtube search).for each question like this respond with (y/n) accordingly.



VI. SETTING UP OWN COMMANDS

We can verify speech to text by using the command `./speech-recog.sh` in the directory `/home/pi/PiAUISuite/voice` command. this software is run continuously using the command `sudo voicecommand -c` in the terminal. by default the keyword is "pi" to activate. you can change the keyword and other commands by opening the command configuration file. this is done by entering the command `voicecommand -e`. by this we can find no of keywords and commands. in this each command is linked to particular task or action. For an example `youtube==youtube-search...` when it records the voice as youtube apple, it does the youtube search for apple.

So if you want to add new voice command just type the command and its action. for an example for the command `check internet` that pings the google for checking the internet connectivity. then in configuration command file type `check internet==ping google.com`. thus it pings when you say check internet.

Since raspberry pi can be used for robotics and many other useful projects therefore voice assistant can be implemented to the robots controlled by microcontroller like raspberry pi.

VII. GOOGLE ASSISTANT

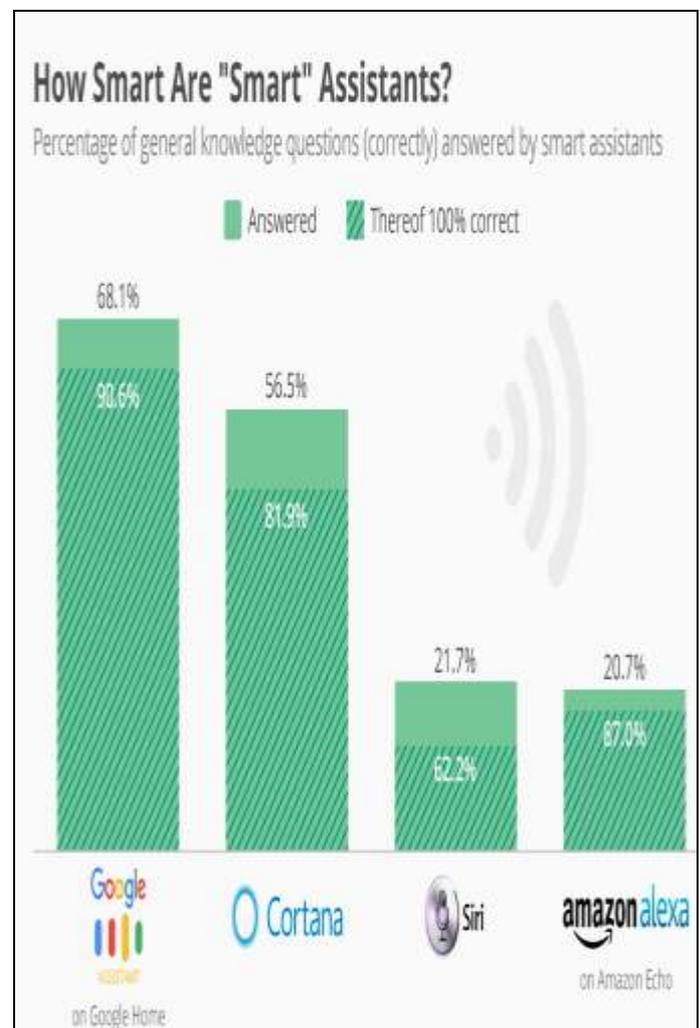
Google Assistant is a virtual personal assistant developed by Google and announced at its developer conference in May 2016. Unlike Google Now, the Google Assistant can engage in two-way conversations.

Assistant initially debuted as part of Google's messaging app Allo, and its voice-activated speaker Google Home. After a period of exclusivity on the Pixel and Pixel XL smartphones, it began to be deployed on other Android devices in February 2017, including third-party smartphones and Android Wear, and was released as a standalone app on the iOS operating system in May. Alongside the announcement of a software development kit in April 2017, the Assistant has been, and is being, further extended to support a large variety of devices, including cars and smart home appliances. The functionality of the Assistant can also be enhanced by third-party developers.



Users primarily interact with the Google Assistant through natural voice, though keyboard input is also supported. In the same nature and manner as Google Now, the Assistant is able to search the Internet, schedule events and alarms, adjust hardware settings on the user's device, and show information from the user's Google account. Google has also announced that the Assistant will be able to identify objects and gather visual information through the device's camera, and support purchasing products and sending money.

The Google Assistant library for Python is a turnkey solution for anyone who wants to quickly integrate the Assistant into a prototype device. The library is written in Python and is supported on popular prototyping devices such as the Raspberry Pi 3. Further Amazon has also introduced their own voice assistant "Alexa", Microsoft's "Cortana" and Apple well known "Siri". All of them are equally efficient and strong but the most favourable is Google Now, the Google Assistant. All of these voice assistants have their own wake up command after which they link with the database or cloud data and reply in a very efficient way. According to the survey by Statista the following data has been obtained:



VIII. CONCLUSIONS

This software system using raspberry pi is a voice recognition software using the Google API's. It can be used in many ways. It also can be used in home automation system and any other interactive systems. It uses webcam or an microphone of USB type connected to raspberry pi to record the voice. Voice recognition is mainly done by using the Google API's as it converts voice to text and the particular task is done for the specific commands. The coding is done using python. We can give different tasks for different command. We can add many commands in command configuration file. We have develop application, in which user can easily send a message with their voice command and also tried to use most of the inbuilt application with voice command. But all these application have adaptation for the English language. we tried to develop application in such a way that it will be useful to the visually impaired persons as the whole of our system work with a voice command. We are also surveying to use mailing and calendar where user will be able to mail and also create a event with the voice command.

The Objective for future development is to develop a voice engine which can also work for the other local language like Hindi.

ACKNOWLEDGEMENT

I am highly indebted to Mr. Neven Nikolov (professor Computer science department, Technical University of Sofia, Bulgaria) for his guidance and constant supervision as well as for providing necessary information.

REFERENCES

1. Yunxin Zhao, A Speaker-Independent Continuous Speech Recognition System Using Continuous Mixture Gaussian. IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING, VOL. 1, NO. 3, JULY 1993.
2. Brandon Ballinger, Cyril Allauzen, Alexander Gruenstein, Johan Schalkwyk, On-Demand Language Model Interpolation for Mobile Speech Input INTERSPEECH 2010, 26-30 September 2010, Makuhari, Chiba, Japan, pp 1812-1815.
3. Ryuichi Nisimura, Jumpei Miyake, Hideki Kawahara and Toshio Irino, —Speech-To-Text Input Method For Web System Using JavaScript, IEEE SLT 2008 pp 209-212.
4. S. Katz. Estimation of probabilities from sparse data for the language model component of a speech recognizer. In IEEE Transactions on Acoustics, Speech and Signal Processing, volume 35, pages 400-401, March 1987.

5. D. Povey, D. Kanevsky, B. Kingsbury, B. Ramabhadran, G. Saon, and K. Visweswariah. Boosted MMI for model and feature-space discriminative training. In Proc. of the Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP), 2008.
6. <http://developer.android.com/index.html>
7. www.stackoverflow.com
8. <http://www.tutorialspoint.com/html/>