Development of a framework for a collaborative and personalised voice assistant

Sangeetha Manoharan* and Parth Natu

Department of Electronics and Communication Engineering,
SRM Institute of Science and Technology,
SRM Nagar, Kattankulathur-603203, Chennai, India
Email: Sangeetm@srmist.edu.in
Email: parth.natu@gmail.com
*Corresponding author

Abstract: Virtual assistant is an artificial intelligence (AI) system that understands natural voice commands and completes the task for the user. There is a need for seamless integration of high level general purpose voice assistants such as Google Assistant and Amazon Alexa under a single framework. In this paper, a novel collaborative personalised voice assistant with the ability to interface hardware components for an interactive environment is proposed. The proposed assistant takes the users' voice input as the query and processes it using a natural language processing (NLP) unit which generates intents from the converted text. Based on the type of intent, the NLP unit passes to one of the two services Google Assistant or Amazon Alexa. If the query is related to requesting information, the NLP passes to Google Assistant which gives an appropriate answer as the voice input and/or a query related to controlling hardware objects, will be pass on to Amazon Alexa. To demonstrate our collaborative voice assistant, a servo motor is used as a hardware object which controls the movement of the 5-inch liquid crystal display (LCD). Results prove that the proposed collaborative voice assistant brings together the strengths of Google Assistant and Amazon Alexa in a single framework.

Keywords: artificial intelligence; natural language processor; recurrent neural network; RNN; voice assistant.

Reference to this paper should be made as follows: Manoharan, S. and Natu, P. (2021) 'Development of a framework for a collaborative and personalised voice assistant', *Electronic Government*, Vol. 17, No. 1, pp.96–104.

Biographical notes: Sangeetha Manoharan has completed her PhD in SRM Institute of Science and Technology in the area of Chaos Communication in 2014. Currently, she is working as an Associate Professor in SRM Institute of Science and Technology. She has published 18 papers in international journals and 15 papers in international conferences. Her research interest includes signal processing applications for wireless communications, internet of things (IoT) and artificial intelligence (AI).

Parth Natu received his BTech in Electronics and Communication Engineering from SRM Institute of Science and Technology in 2018. Currently, he is pursuing his MS in Computer Science in Pennsylvania State University (Penn State). His research interest includes artificial intelligence and machine learning (ML).

Copyright © 2021 Inderscience Enterprises Ltd.

1 Introduction

Nowadays people are surrounded by smart devices like smartphones, smart watches, and internet of things (IoT) devices are giving voice assistants more utility in a connected user's life. When voice assistants began to emerge in 2011 with the introduction of Siri, no one could have predicted that this novelty would become a driver for tech innovation. But now, it's estimated that every one in six Americans own a smart speaker (Google Home, Amazon Echo) and eMarketer forecasts that nearly 100 million Smartphone users will be using voice assistants in 2020 (https://clearbridgemobile.com/7-key-predictions-for-the-future-of-voice-assistants-and-ai/). Many Companies have established Virtual Personal Assistants such as Microsoft's Cortana, Apple's Siri, Facebook's M, Google Assistant and Amazon Alexa. But the most popular virtual assistant platforms are Google Assistant and Amazon Alexa. These are proprietary software operated by independent companies. They function on the knowledge database developed by their respective companies. These databases consist hundreds of thousands of possible queries a user can ask. Since they are maintained by individual companies, their key functionality and behaviour is different. Google Assistant has focused more on query based tasks such as searching for an address or user specific queries whereas Amazon has trained its Alexa in home automation and IoT.

Although, Google Assistant and Amazon Alexa have a huge database for almost every possible query, they have focused on different sets of queries they expect their user to ask. Google Assistant does not provide good hardware integration and IoT capabilities but has good responses for conversation based queries. Amazon Alexa lacks in the accuracy of the answers it provides for the conversation based queries, but the hardware integration support on Alexa is very good. The main motivation of our paper is to develop a collaborative framework that integrates the features of Google Assistant and Alexa, thereby providing an efficient personalised voice assistant. Thus, the main aim of this work is to:

- Integrate both Amazon Alexa and Google Assistant in a single framework using Raspberry Pi.
- Use a hotword engine that constantly listens for the keyword to access the framework.
- Train the assistant to use Google Assistant for conversations and Amazon Alexa for hardware integration.
- Provide hardware support to make the assistant more interactive.

The rest of this paper is organised as follows: in Section 2 related works are presented. Section 3, discuses briefly the system framework. Software model description and algorithms for data flow are given in Section 4. Results and discussions are presented in Section 5 and concluding remarks are given in Section 6.

2 Motivation and related works

Smart environments can be designed using artificial intelligence (AI) and machine learning (ML) to understand the environment itself and the needs of its inhabitants.

Security for the virtual assistants through facial recognition is proposed in Praddeep et al. (2019). Academic institutions are trying to enhance their campuses to smart campus by providing students with novel smart services that can make everyday activities easier.

In 2019, Gaglio et al. presented a virtual assistant using IBM Watson Technology to assist the staffs and the students of smart campus at the University of Palermo. The basic functioning of voice assistants, its constraints are analysed in Polyakov et al. (2018) and the authors also developed an intelligent voice assistant for IOT devices using ML. The performance of various NLP systems used by Google Assistant and Alexa is evaluated in (https://medium.com/snips-ai/benchmarking-natural-language-understanding-systems-google-facebook-microsoft-and-snips). The performance of these assistants are benchmarked on the basis of intents like: get weather, play music, book restaurant, search item, add to playlist, rate book, search movie schedule etc., .Comparison chart showing the percentage of accuracy, on the response provided these assistants towards certain type of query is shown.

A multi modal dialogue systems processing two or more combined user input modes such as speech, image, video, touch, manual gestures is presented in Këpuska and Bohouta (2018). This new model of assistant uses different technologies such as gesture recognition, image/video recognition, speech recognition, the vast dialogue and conversational knowledge database, so as to increase the interaction between humans and machines. Performance of the voice assistants can be improved by learning mechanism to acquire new information on user behaviour. In Mane et al. (2017) ML-based smart personal assistant is proposed and they have also used interrupt based broadcast receiver approach for less power consumption.

A detailed analysis of the internal hardware components and implementation of Google Assistant is provided in Dempsey (2017). The hardware contains a main processing board connected to the Internet, a far field microphone array for voice input, speakers to emit a response and a mechanical chassis to hold these components together. Similarly, in Amazon Alexa, the hardware descriptions are given in Dempsey (2015).

The usefulness and the timely information from intelligent personal assistants (IPAs) is presented in Goksel-Canbek and Mutlu (2016). Use of these assistants built into the mobile operating systems so that the users can perform their daily tasks such as taking dictation, getting turn-by-turn directions, vocalising email messages, reminding daily appointments, setting reminders or responding any factual questions. Few popular IPAs are Apple's Siri, Google Now and Microsoft Cortana.

Development of home based virtual assistant called JIBO is presented in Rane et al. (2014), and it elaborates the technical specification of the hardware and the software used in JIBO. The paper also analysed the limitations and the areas for improvement in JIBO.

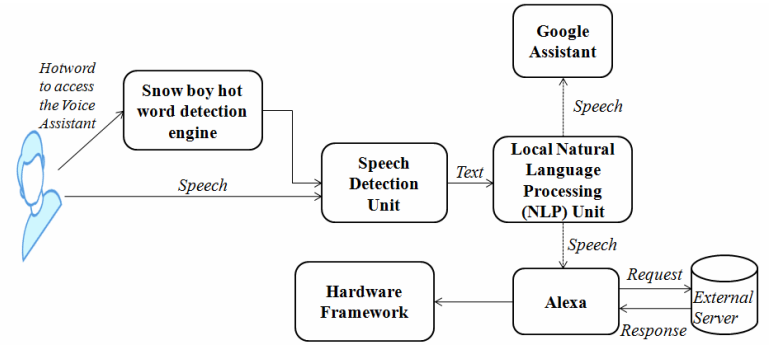
Graves et al. in 2013 explained the process of speech recognition using recurrent neural network (RNNs). They used the process of splitting audio input into its constituent alphabets and transcribing them to individual words, for speech recognition. These RNNs helps in building a precise virtual assistant.

3 System framework

3.1 Framework description

The proposed collaborative and personalised voice assistant is shown in Figure 1.

Figure 1 Proposed collaborative and personalised voice assistant framework (see online version for colours)



The framework consists of a hotword detection engine, speech detection unit, local natural language processing (NLP) unit, Google Assistant and an Alexa. Snowboy is an highly customisable hotword detection engine that is embedded real-time and is always listening (even when off-line) compatible with Raspberry Pi, (Ubuntu) Linux, and Mac OS X.A hotword is a keyword or phrase that the computer constantly listens for, as a signal to trigger other actions. The Snowboy Library is used for hotword detection (http://docs.kitt.ai/snowboy/#introduction). In this work, the hotword used is 'Snowboy' and it is highly customisable, allowing the users to freely define their own magic hotword. The Hotword Detection engine loops every 30 millisecond to check for the hotword 'Snowboy'. Once the hotword is detected, the framework will then listens for the following query.

The speech detection unit is used to convert the speech signal into text information that can be processed by a Local NLP unit. The Local NLP Unit is responsible for interpreting the speech in the form of text and decides whether to use the Google Assistant or Amazon Alexa based on the query. The Local NLP unit is a naive in nature. It looks for particular keywords in the spoken phrase to trigger hardware interaction through Amazon Alexa. If it does not detect, it assumes that the query is knowledge based query and passes on to the Google Assistant.

The Software model of the proposed frame work has two new blocks as compared to the existing voice assistant system. Th first blocks being the decision system that decides upon which service to use-Google assistant or the Amazon Alexa, based on the query supplied to it which is stored offline in a Raspberry Pi, the main processor board used in the proposed system. The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV. It is a capable little device that enables all kind of computing tasks (https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/).

The second block is a framework for incorporating hardware components such as cameras, robotic arms, displays and sensors, with the voice assistant, for an interactive environment. The hardware part of the proposed frame work is designed such that, it completely demonstrates the software capabilities provided at the backend.

In this work, the hardware framework consists of a face recognition robot having a two degree of freedom pan tilt platform (PTP) that makes it possible for the camera to point in a desired. The robot is controlled by a servo motor. Raspberry Pi 3 is. It is used to store and execute the software stack. Various Libraries, the source code and the main

operating system are stored in the Raspberry Pi. It connects to the internet via Wi-Fi to perform remote procedure calls (RPC).

3.2 Hardware and software requirements

Table 1 Hardware specifications and justifications

Hardware requirements	*Specification*	*Justification*
Actuators	MG995 servo motors	For pan/tilt
Arduino	Arduino ATMega 328P	For controlling servos
Touch display	Waveshare 5-inch HDMI LCD	For screen interaction
Camera with microphone array	PS3 eye camera	For face and speech detection
Speaker	JBL Go	For voice output
Processor board	Raspberry Pi 3 Model B	Microcontroller the main processing board

Table 2 Software specifications and justifications

Software	*Specification*	*Justification*
Robot operating system (ROS)	ROS Indigo	ROS simplifies the communication between each part of the software stack.
Google Assistant gRPC API	google-assistant-grpc 0.1.0	gRPC API allows the software stack to make calls to the Google Assistant service online by sending audio requests by the user and receiving an audio response
Alexa Skills API	---	The Alexa Skills API parses text input to give an output of intents which can be used to trigger hardware components of the virtual assistant
Nanpy Python Library	nanpy 0.9.6	The Nanpy library allows the Raspberry Pi to make a serial connection with boards such as an Arduino to actuate electronics such as servos
Snowboy Hotword Detection Engine	v1.1.1 (2017-03-24)	Snowboy is a hotword detection engine uses library to listen for a custom hotword and perform an action after it is detected

4 Methodology for data flow in the virtual assistant

The procedure adopted for data flow in the virtual assistant is as follows:

Step 1 Start the Voice Assistant with the user pronouncing the hotword. In this work, the hotword used is ‘Snowboy’.

Step 2 Once the hotword is detected, the assistant continue to listen to the query.

Step 3 Robot operating system (ROS) is used to handle user queries and pass them seamlessly to the speech detection unit.

Step 4 The local NLP unit uses Python and DialogFlow for intent detection and detection.

Step 5 A decision unit is used to decide upon which assistant to use, either the Google Assistant or the Amazon Alexa.

Step 6 If the assistant chooses Google Assistant, then google-assistant-grpc 0.1.0 allows the software stack to make calls to the Google Assistant service online by sending audio requests by the user and receiving an audio response.

Step 7 If the assistant chooses Amazon Alexa, Alexa Voice Services (AVS) allows us to get connected to the voice enabled products such as microphone and speaker.

4.1 Hotword detection algorithms

A hotword is an uttered word that the microphone keeps listening for, to perform a certain action. The popular hotword includes 'Alexa' on Amazon Echo (https://developer.amazon.com/alexa), 'OK Google' on Google Assistant (https://assistant.google.com/intl/en_in/) and 'Hey Siri' on iPhones (https://www.apple.com/in/ios/siri/). In our work, 'Snowboy' is the hotword used. The algorithm explaining how a hotward detection works is given as follows:

Algorithm 1 Hotword detection

1	Import Libraries for Hotword Detection
2	Import .umdl file for particular hotword (e.g. snowboy.umdl)
3	Setup hotword detection loop
4	Loop through ring buffer every 0.03 seconds till microphone data matches
5	if hotword detected, run speech detector program

4.2 Speech detection algorithm

The speech detection unit uses a speech detection algorithm, where the speech signal is sampled at the rate of 16 KHz to output a matrix of amplitude values at every 1/16000th of a second. This matrix cannot be directly fed into a neural network for detection because it has number of data with which it has to compare for an accurate classification and processing the speech signal in frequency domain gives more accurate results. So, the matrix is divided into smaller chunks, each chunk is converted into frequency domain representation and then fed to a RNN for alphabet classification. The speech detection algorithm is given as follows:

Algorithm 2 Speech detection

1	Import audio capture libraries
2	Capture audio and sample at 16 KHz
3	Store sampled data in an array
4	Divide the array into 20 ms chunks
5	Apply Fourier transform on to the array to convert it to frequency domain
6	Feed this data to a RNN, trained for speech to text conversion

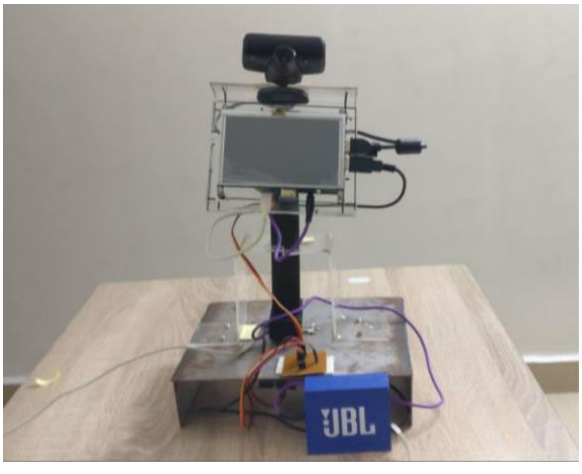
7 Extract classified text into an array
8 Post-process the text to remove repetitions and gaps
9 Publish the output as text

The speech signals cannot be defined in a single time interval, the individual letters may be extended due to the pronunciation, background noise and hardware malfunctions. Therefore, the RRN output is post processed to remove any repeated characters and gaps to give a proper output.

5 Results and discussion

In this section we show some of the results achieved out of the proposed collaborative and personalised voice assistant. Figure 2 shows the developed voice assistant

Figure 2 Collaborative and personalised voice assistant (see online version for colours)

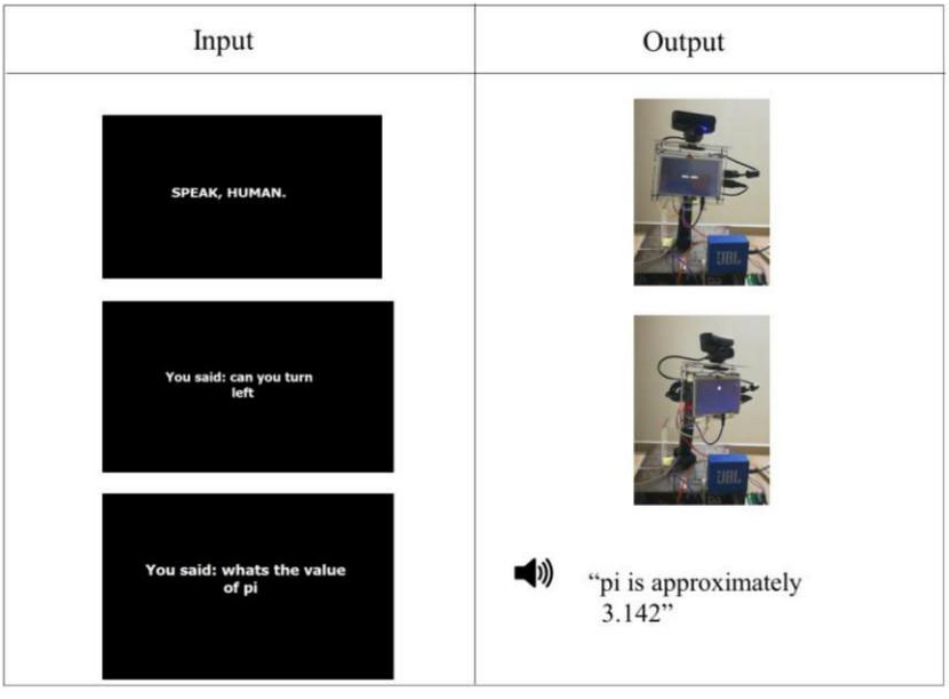


The user speaks out the hotword 'Snowboy', once it is recognised, the voice assistant shows a cue saying 'SPEAK, HUMAN' on its 5-inch LCD display. The user then instructs to turn left by saying 'Can you turn left'. The voice assistant displays the same and takes decision upon which assistant to use. As the instruction belongs to controlling/movement of the hardware, the proposed assistant used Alexa to execute the same. Now the user asks for the value of π, which is a knowledge based question, so the assistant chooses Google Assistant to answer for the query.

The proposed personalised voice assistant is capable of answering different types of questions, using Google Assistant or the Amazon Alexa. Figure 3 shows an example of conversation.

The intent behind the user's input is recognised and the assistant is capable of making decision on which assistant to use and immediately respond to the same. While the second input requires the Google assistant to reply for the query and the answer is immediately displayed on the 5-inch LCD display unit.

Figure 3 Collaborative personalised voice assistant conversation in input/output form (see online version for colours)



6 Conclusions

Nowadays, virtual assistants can be found on every smart device and are constantly used by people for a variety of purposes. In this work we presented a collaborative framework for personalised voice assistant that integrates the strengths of Google Assistant and Amazon Alexa. To this aim, a conversational assistant, capable of making decision upon which database to use and answering to the questions or control the movement of hardware components that can be integrated with the assistant. The proposed framework can be accessed via a Hotword Detection Engine. The major contribution in the proposed work lies in the training of assistant to use Google Assistant for conversations, knowledge based queries and Amazon Alexa for controlling hardware objects. Finally, the virtual assistant functionalities could be extended to provide more interactive environment by integrating more and needed hardware components in the hardware framework. This kind of collaborative and personalised voice assistants can be used in administrative offices, smart home automation, smart hotel management and smart office integrations.

References

Dempsey, P. (2015) 'The teardown Amazon echo digital personal assistant [teardown consumer electronics]', *Engineering and Technology*, March, Vol. 10, No. 2, pp.88–89.

Dempsey, P. (2017) 'The teardown: Google Home personal assistant', *Engineering and Technology*, April, Vol. 12, No. 3, pp.80–81.

Gaglio, S., Re, G.L., Morana, M. and Ruocco, C. (2019) 'Smart assistance for students and people living in a campus', *Proceedings of 2019 IEEE International Conference on Smart Computing (SMARTCOMP)*, Washington, DC, USA, 12–15 June.

Goksel-Canbek, N. and Mutlu, M.E. (2016) 'On the track of artificial intelligence: learning with intelligent personal assistants', *Journal of Human Sciences*, January, Vol. 13, No. 1, pp.592–601.

Graves, A., Mohamed, A-R. and Hinton, G. (2013) 'Speech recognition with deep recurrent neural network', *Proceeding of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2013)*, Vancouver, BC, Canada, 26–31 May.

Këpuska, V. and Bohouta, G. (2018) 'Next-generation of virtual personal assistants (Microsoft Cortana, Apple Siri, Amazon Alexa and Google Home)', *Proceedings of the 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, USA, 8–10 January.

Mane, P., Sonone, S., Gaikwad, N. and Ramteke, J. (2017) 'Smart personal assistant using machine learning', *Proceedings of the 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, Chennai, India, 1–2 August.

Polyakov, E.V., Mazhanov, M.S., Rolich, A.Y., Voskov, L.S. et al. (2018) 'Investigation and development of the intelligent voice assistant for the internet of things using machine learning', *Proceedings of 2018 Moscow Workshop on Electronic and Networking Technologies (MWENT)*, Moscow, Russia, 14-–16 March.

Praddeep, P., Balaji, P. and Bhanumathi, S. (2019) 'Artificial intelligence based person identification virtual assistant', *International Journal of Recent Technology and Engineering (IJRTE)* , September, Vol. 8, No. 2S11, pp.2315–2319.

Rane, P., Mhatre, V. and Kurup, L. (2014) 'Study of a home robot: JIBO', *International Journal of Engineering Research and Technology*, October, Vol. 3, No. 10, pp.490–493.

Websites

http://docs.kitt.ai/snowboy/#introduction (accessed 7 November 2019).

https://assistant.google.com/intl/en_in/ (accessed 7 November 2019).

https://clearbridgemobile.com/7-key-predictions-for-the-future-of-voice-assistants-and-ai/ (accessed 18 July 2019).

https://developer.amazon.com/alexa (accessed 7 November 2019).

https://medium.com/snips-ai/benchmarking-natural-language-understanding-systems-google-facebook-microsoft-and-snips (accessed January 2018).

https://www.apple.com/in/ios/siri/ (accessed 7 November 2019).

https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/ (accessed 7 November 2019).