

Advanced Virtual Assistant Based on Speech Processing Oriented Technology on Edge Concept (S.P.O.T)

Akhilesh L

PG Scholar, D.C.S., Cochin University of Science and Technology, Kochi, Kerala, India

ABSTRACT

With the advancement of technology, the need for a virtual assistant is increasing tremendously. The development of virtual assistants is booming on all platforms. Cortana, Siri are some of the best examples for virtual assistants. We focus on improving the efficiency of virtual assistant by reducing the response time for a particular action. The primary development criterion of any virtual assistant is by developing a simple U.I. for assistant in all platforms and core functioning in the backend so that it could perform well in multi platformed or cross platformed manner by applying the backend code for all the platforms. We try a different research approach in this paper. That is, we give computation and processing power to edge devices itself. So that it could perform well by doing actions in a short period, think about the normal working of a typical virtual assistant. That is taking command from the user, transfer that command to the backend server, analyze it on the server, transfer back the action or result to the end-user and finally get a response; if we could do all this thing in a single machine itself, the response time will get reduced to a considerable amount. In this paper, we will develop a new algorithm by keeping a local database for speech recognition and creating various helpful functions to do particular action on the end device.

KEYWORDS: Virtual Assistant, Speech processing, SPOT, Text to Speech, Music player

INTRODUCTION

SPOT (Speech Processing Oriented Technology) is computer software that we designed to help the user to take the whole control of a computer using a speech. Thus SPOT replaces peripherals for your computer. It acts as an emulator on your WINDOWS device, which replaces the keyboard and general G.U.I. Buttons to trigger actions. SPOT consists of modules that can perform many actions simply by giving the Speech as input. Here the software accepts inputs from the user, and the computer application maps the user input into corresponding hardware input. Thus computer application acts as a driver for controlling the peripherals and other software on the computer. The software installed on the computer will take control of the microphone, and the speaker who is currently active and further processing is done based on setting these as default devices. The primary development criterion of any virtual assistant is by developing a simple U.I. for assistant in all platforms and core functioning in the backend so that it could perform well in multi platformed or cross platformed manner by applying the backend code for all the platforms. We try a different research approach in this paper. That is, we give computation and processing power to edge devices itself. So that it could perform well by doing actions in a short period, think about the normal working of a typical virtual assistant. That is taking command from the user, transfer that command to the backend server, analyze it on the server, transfer back the action or result to the end-user and finally get a response, if we could do all this thing in a single machine itself, the response time will get reduced to a considerable amount. In this research, we will develop a new algorithm by

How to cite this paper: Akhilesh L "Advanced Virtual Assistant Based on Speech Processing Oriented Technology on Edge Concept (S.P.O.T)" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-4 | Issue-6, October 2020, pp.159-162, URL: www.ijtsrd.com/papers/ijtsrd33289.pdf



Copyright © 2020 by author(s) and International Journal of Trend in Scientific Research and Development Journal. This is an Open Access article distributed under the terms of the Creative Commons Attribution License (CC BY 4.0) (<http://creativecommons.org/licenses/by/4.0>)



keeping a local database for speech recognition and creating various helpful functions to do particular action on the end device. The development of virtual assistants is booming on all platforms. Cortana, Siri are some of the best examples for virtual assistants. We focus on improving the efficiency of virtual assistant by reducing the response time for a particular action. After implementing the system, we also try to compare it with existing systems so that we could evaluate how successful our research is and how much contribution we have given to upcoming ideas.

MODULARIZATION

Our proposed system is designed in such a way that separate modules are integrated together to form a single system. Each module is developed by maintaining the encapsulation feature of the object-oriented programming concept.

A. Speech To Text Module

Speech to text module is one of the core modules of our proposed system. There are many API available for Speech to text conversion. Instead of using those API's We build our own speech recognition algorithm by using the speech recognition engine feature of Microsoft visual studio. The basic idea is to build a dictionary of words so that the created speech recognition engine could find the exact word without much delay. We use delimiters in a format that the first word represents the word has to be recognized by the system; the second word describes the

word that passed to the text to speech module to generate Speech, which the system has to talk. The dictionary that we have built can be kept as a text file in any format instead of using complex database structures so that a user can update or add more words to the dictionary in the future. On recognition of the exact word from the dictionary, the system checks the third word after the second delimiter, and the third word means whether it is a command or just a greeting from the author to the computer. The third word could be yes or no; if it is yes, then the system checks the word to what action should be performed. If it is NO, then the system sends the second word to the speech generation module to get speech output from the system. For the Speech to text module, the necessary criteria are It should have a microphone by default or an external microphone should be connected. This is the concept of text to speech module of our virtual assistant. The main objective of making a local database is the same as we discussed in the introduction section, which is to make the system work faster by avoiding a server. If we need, we can add API for getting a better response if the third word after the second delimiter is NO, Which means it is not a command or not an immediate action that the system can perform. So we will get a better response, but it takes some time for that; at the same time, if it is an immediate action that is the word is YES, We will surely get an immediate response without waiting for the API.

B. Information retrieval module

Here we use web scrapping from Wikipedia by means of passing search words to Wikipedia and retrieve data related to the search query, as shown in fig2. When the user has some doubt in some topics while learning, the user can ask the system for the details by activating teaching mode by the keyword "what?". In this section, Web scrapping Wikipedia does the data fetching, and after splitting the fetched data as per the user needs, the data will send to the text to the speech section, which will read out the data to the user. The main aim of this module is to provide information regarding anything with the help of Wikipedia, and this is the essential online component of our system.

C. Hand free Desktop Access Module

The core feature of this module is that even a paralyzed user can access the pc just by giving speech input. The hand-free access controls include the opening of installed applications, the start of my computer & files, playing audio with default players, copy files, cut files, shutdown pc, locking pc, aborting shutdown, opening and seeing photos in pc, doing arithmetic problems, searching video in YouTube, search image and web pages in Google, etc.

D. Text Editor Module

This section controls the main actions like select, copy, cut, paste, etc. in real-time, which enables the user to perform these tasks by merely giving the command as Speech. The concept of hands free text editor mode is evolved from the integration of speech recognition and virtual keyboard. In the

virtual keyboard, a key press is simulated by G.U.I. That is when we click on respective G.U.I. Buttons using mouse clicks corresponding send keys method sends a key press to the system. We use that concept by avoiding button click and integrate the idea with speech recognition. That is, if we tell the system to generate key press, it will invoke send key method and simulate key press.

E. Music Player Module

The music player module imports almost all the properties of a windows media player so that the application can play songs without any pre-installed media player, and everything is controlled with Speech. The path for the music folder has to be set by the user. Once the object for the music player is created, it will be assigned the path and start playing the music

F. CD Drive Handling Module

In this section, a user can handle the cd drive, which is the hardware component of the system, just by giving the Speech commands. Here we create a handle to access the cd drive and using that designed handle the application can eject the cd drive which is triggered the speech input given by the user

G. Video Player Handling Module

Video player module includes speech-based control for a default media player, which can play, pause, mute, unmute, adjust the screen size, show subtitles, seek forward and backward, etc. only by some speech commands

SYSTEM DESIGN

Our proposed system is designed in such a way that separate modules are integrated together to form a single system. Each module is developed by maintaining the encapsulation feature of the object-oriented programming concept. Since the Speech is given as input, even a paralyzed person can control the system by installing and giving his Speech as input to our application in this system. We keep track of various commands in a notepad file, which is easy to access by the software and thereby fastening the process. The chat bot system will work in such a way that by using the concept of A.I.M.L. So that the chatbot system may take some more time to get a response since there is a vast database to gain access. Our key idea is that we keep a small database for doing particular tasks. So it will help the system to perform faster

PERFORMANCE EVALUATION

After implementing the proposed system, we have to evaluate the performance of the implemented system so that we can say our project is a great success, and we could give a new concept or new method for the upcoming projects, which could have a huge impact in the near future. Performance evaluation is done by calculating the time for getting a response to do particular action to get performed on the system. So the time to perform a series of events is calculated and find the average time gives the average response time of the system.

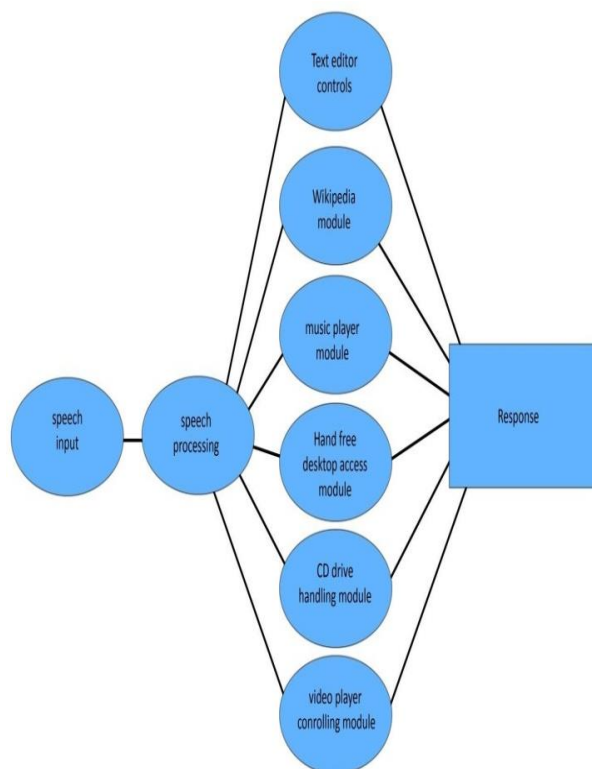


Fig.1

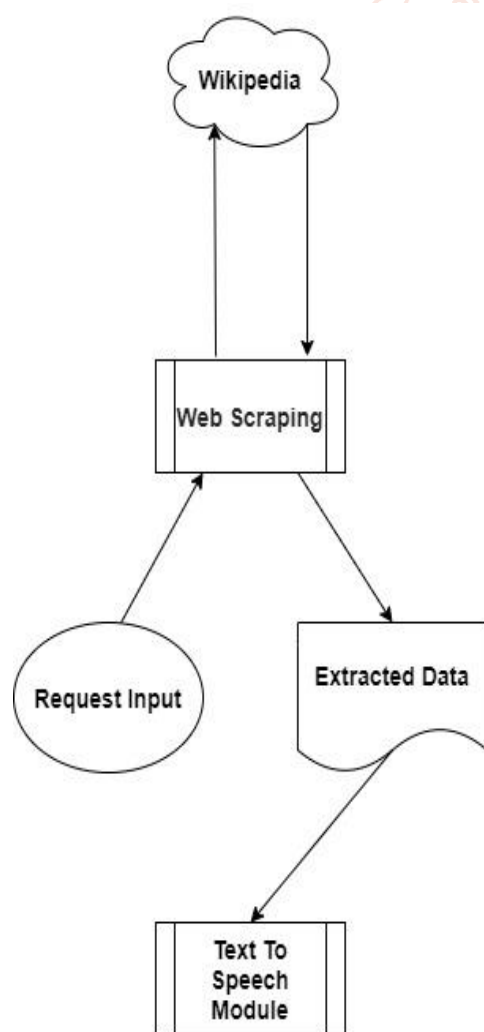


Fig.2

ACHIEVEMENTS

Comparing to the existing systems, the software SPOT provides high flexibility, and the system gives a better

operating environment for users by providing all in one P.C. Customizable controllers make the system more user friendly along with the Implementation of the system, which works in both online and offline modes and easy to use.

CONCLUSION

The software SPOT provides a much more comfortable and flexible system control that replaces G.U.I. Based controls. The development of the software has gone through various stages, like System analysis, System design, System Testing, and Implementation. Our system provides exclusive options for individual users to set their settings for their comfort. I think that the hard work gone behind the development of the system has been fruitful.

FUTURE WORK

In this paper, we focus more on improving the response time of the system; in the future, we focus on the auto-learning system for particular events, which makes the system much smarter rather than faster.

ACKNOWLEDGEMENT

First of all, I thank God for helping me to complete the project and leading me to success. I Sincerely thank H.O.D., My teachers, and my friend Jeenamol Abraham for motivating me in every step of my project. I also thank anonymous reviewers who put their effort into my project to get published.

REFERENCES

- [1] X. Lei, G. Tu, A. X. Liu, C. Li and T. Xie, "The insecurity of home digital voice assistants - amazon alexa as a case study", *CoRR*, vol. abs/1712.03327, 2017.
- [2] J. Gratch, N. Wang, J. Gerten, E. Fast and R. Duffy, "Creating rapport with virtual agents" in *Intelligent Virtual Agents*, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 125-138, 2007.
- [3] J. C. Lester, S. A. Converse, S. E. Kahler, S. T. Barlow, B. A. Stone and R. S. Bhogal, "The persona effect: Affective impact of animated pedagogical agents", *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems C.H.I.* '97, pp. 359-366, 1997.
- [4] E. André and C. Pelachaud, *Interacting with Embodied Conversational Agents*, Boston, MA: Springer US, pp. 123-149, 2010.
- [5] B. Weiss, I. Wechsung, C. Kühnel and S. Möller, "Evaluating embodied conversational agents in multimodal interfaces", *Computational Cognitive Science*, vol. 1, pp. 6, Aug 2015.
- [6] Matsuyama, A. Bhardwaj, R. Zhao, O. Romeo, S. Akoju and J. Cassell, "Socially-aware animated intelligent personal assistant agent", *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pp. 224-227, 2016.
- [7] M. Schroeder, E. Bevacqua, R. Cowie, F. Eyben, H. Gunes, D. Heylen, et al., "Building autonomous sensitive artificial listeners", *IEEE transactions on affective computing*, vol. 3, pp. 165-183, 2012.
- [8] F. Battaglia, G. Iannizzotto and L. Lo Bello, "A person authentication system based on rfid tags and a cascade of face recognition algorithms", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, pp.

1676-1690, Aug 2017.

- [9] F. Schroff, D. Kalenichenko and J. Philbin, "Facenet: A unified embedding for face recognition and clustering", *2015 IEEE Conference on Computer Vision and Pattern Recognition (C.V.P.R.)*, pp. 815-823, June 2015.
- [10] M. C. Gombolay, R. A. Gutierrez, S. G. Clarke, G. F. Sturla and J. A. Shah, "Decision-making authority team efficiency and human worker satisfaction in mixed human-robot teams", *Autonomous Robots*, vol. 39, no. 3, pp. 293-312, 2015.
- [11] L. Huang, L.-P. Morency and J. Gratch, "Virtual rapport 2.0", *International Workshop on Intelligent Virtual Agents*, pp. 68-79, 2011.
- [12] T. Bickmore and D. Schulman, "A virtual laboratory for studying longterm relationships between humans and virtual agents", *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems*, vol. 1, pp. 297-304, 2009.
- [13] M. Harbers, J. M. Bradshaw, M. Johnson, P. Feltoich, K. van den Bosch and J.-J. Meyer, "Explanation in human-agent teamwork" in *Coordination Organizations Institutions and Norms in Agent System VII*, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 21-37, 2012.
- [14] T. Bagosi, K. V. Hindriks and M. A. Neerincx, "Ontological reasoning for human-robot teaming in search and rescue missions", *The Eleventh ACM/IEEE International Conference on Human Robot Interaction*, pp. 595-596, 2016.
- [15] L. López, S. Martínez-Fernández, C. Gómez, M. Choraś, R. Kozik, L. Guzmán, et al., "Q-rapids tool prototype: supporting decision-makers in managing quality in rapid software development", *International Conference on Advanced Information Systems Engineering*, pp. 200-208, 2018.
- [16] S. Dueñas, V. Cosentino, G. Robles and J. M. Gonzalez-Barahona, "Perceval: Software project data at your will", *Proceedings of the 40th International Conference on Software Engineering: Companion Proceedings*, pp. 1-4, 2018.
- [17] E. Souza, R. Falbo and N. L. Vijaykumar, "Using ontology patterns for building a reference software testing ontology", *2013 17th IEEE International Enterprise Distributed Object Computing Conference Workshops*, pp. 21-30, 2013.

