Voice-Enabled Virtual Assistant

Ch. Lakshmi Chandana, V. Ashita, G. Neha, K. Sravan Kumar, D. Suresh Babu, G. Krishna Kishore, and Y. Vijaya Bharathi

Abstract Frameworks are being learned step by step nowadays and will assist people in their daily lives. Moreover, artificial intelligence [AI] techniques are now available in a broad number of sectors, ranging from ventures in manufacturing to medical innovation. As a result, within the completed framework, this research work has created a virtual assistant to solve college-related queries. The resulting framework is essentially a virtual assistant, who is meticulously college organized and resolve all college-related questions for students, teachers, and administration. Students confront a lack of critical information about college difficulties for a variety of reasons. The reason might be a result of system flaws. Such causes include a communication gap between the student and the college administration, a lack of student engagement, a lack of suitable guidance, and ignorance on the part of the student and/or the college administration. The student may be unaware of the class schedule, event timings, event location, vacations, examination schedule, permissions, and placement details. Here, the virtual assistant plays the key role in providing the necessary information. The advancements like natural language processing are utilized with the help of GTTS, Google API to change over from text to speech and the reverse way around. Here, the whole input is taken by the user and driven via graphical user interface (GUI). This application reduces the time and manual work of the user.

Keywords Natural language processing · GTTS · Artificial intelligence · Virtual assistant · Frameworks

Ch. Lakshmi Chandana (✉) · V. Ashita · G. Neha · K. Sravan Kumar · D. Suresh Babu · G. Krishna Kishore · Y. Vijaya Bharathi
Department of CSE, VR Siddhartha Engineering College, Vijayawada, India

G. Krishna Kishore
e-mail: gkk@vrsiddhartha.ac.in

Department of IT, VR Siddhartha Engineering College, Vijayawada, India

P. Karrupusamy et al. (eds.), *Sustainable Communication Networks and Application*,
Lecture Notes on Data Engineering and Communications Technologies 93,
https://doi.org/10.1007/978-981-16-6605-6_24

1 Introduction

Organizations today can give quick and customized reactions to clients with technology. In today's world, people are surrounded by smart watches, smart devices, and IoT devices in most cases. The IoT devices in today's world are capable of interacting with one another to provide useful services without interaction of humans in between. In this developing innovation, savvy conditions can be created utilizing artificial intelligence and machine learning to comprehend the climate needs. Knowing about the college before getting admission is very important. To be aware of the college and its schedules and its placement activities is necessary. To know more about the activities going on in our college without stress, we can use the latest technology. Technology nowadays can recognize our speech and text and can even reply back without human interference. We have seen voice assistants like Google Voice Assistants to know about nearby restaurants, schools, colleges, etc. Students going to college may forget the examination date and schedules; to help the students, the voice assistant is a great technology with the natural language processing technology in nowadays compared to printed ones. Understudies face absence of information on essential data with respect to school issues because of a few reasons. Such reasons incorporate communication holes among understudies and college organization. The understudy may have absence of data about the class timetable, occasion timings, occasions, examination timetable. Here, the virtual assistant assumes the critical part in giving the vital data. Today, the virtual bots are fueled by computerized reasoning; they are consequently, fit for helping through text or voice. Menial helpers give the best answers for both the students and administrators in different regions. Voice-actuated chatbots are the ones who can cooperate and convey through voice. They are equipped for tolerating the order in an oral or composed structure through natural language processing (NLP) innovation. They are customized to answer through voice. Here, this research work enhances the chatbot to furnish clients with the data in regard to college arrangement, class timetable, occasion timings, occasions, examination timetable, and fee structure. This voice-empowered chatbot is intended to furnish clients effortlessly gets to sites and applications.

Examples of our project are the Google Assistant and an Amazon Alexa which perform as according to our voice commands; similarly our application also works according to the user commands but with limitations, as these are restricted to the college website.

The objectives of the project are:

- To foster a remote helper for understudies to address college-based inquiries and give college-related data.
- To design a prototype with efficient performance.
- To develop/implement a working application for designed prototype
- To test the proposed model functioning and its performance on the college website.

Title justification:

The title to this project is "Voice enabled Virtual Assistant"; the virtual is having a meaning of not physically existing as such but made by software to appear to do the task. The virtual assistant is a non-physically existing human type support that is given to make things easy to the human. The title "voice enabled virtual assistant" is a physically non existing body done by software with voice recognition and talk back capability. The voice-enabled virtual assistant is built to help the students to know their schedule and timetable without much stress.

2 Related Work

Siddesh et al. [1] has done the work on "Artificial Intelligence-based Voice Assistant." In this work, the voice assistant performs mental tasks like turning on/off smart phone applications with the help of voice user interface (VUI) which is used to listen and process audio commands. They have installed GTTS (Google Text-to-Speech) engine package to make the voice assistant speak like a normal human being.

Gaglio et al. [2] proposed a method as "Artificial intelligent college oriented virtual assistant." In this work, an app for individual user with individual login and storage of data is developed. The artificial intelligence is used to create a virtual assistant, and the data related to specific user is sored into the application. Each and every individual can have the college data like timetable and examination timings, holidays etc. We can also get the information regarding location of the college and date and time etc.; here, technologies like artificial intelligence and Android studio are used.

Prajwal et al. [3] proposed "Universal Semantic Web Assistant based on Sequence to Sequence Model and Natural Language Understanding." The Web-based virtual assistant named as "Wisdom" is intended to build it is a complex system that includes several modules that are intended to perform specific tasks. These modules are later integrated to form the complex system as a whole. Each of our sub-systems interact to each other through cross module function calls similar to the request and response API's.

Pal et al. [4] proposed an user experience with smart voice assistants: The accent perspective answers the research questions regarding the usability, usage patterns, and the overall satisfaction received after using the VA's by the secondary English language speakers we wield a dual prong strategy: questionnaire survey via Google Forms and evaluating the serviceability of the VA's in a free-living condition. The questionnaire is created elicited from the System Usability Scale. The objective behind the questionnaire survey is of two-fold: (1) to let the users allocate their feedback towards their usability, usage pattern, and the usefulness of the VA's and (2) to act as an input to the second part of the experiment.

Atzeni et al. [5] proposed an "AskCO: A Multi-Language and Extensible Smart Virtual Assistant." It allows answering complex natural language queries by translating them into a proper formal meaning representation. The AskCO can be used to translate natural language utterances into a Java source code.

Bohouta et al. [6] proposed "Next-Generation of Virtual Personal Assistants (Microsoft Cortana, Apple Siri, Amazon, Alexa and Google Home)." It process two or more combined user input modes, such as speech, image, video, touch, manual gestures, gaze, and head and body movement in order to design the next-generation of VPAs model. It can be used in other different areas of application, including education assistance, medical assistance, robotics and vehicles, disabilities systems, home automation, and security access control.

Novikov et al. [7] proposed "Development of Intelligent Virtual Assistant For Software Testing Team." In this work, the team builds an intelligent virtual assistant where one needs to combine in one project different technologies and tools from different areas. User input understanding: audio speech recognition, natural language understanding, user face recognition, sentiment analysis, opinion mining, etc. Intelligent reasoning: context understanding, dialogue management, social reasoning, domain specific knowledge, user model, etc., and next output generation, and software testing.

Iannizzotto et al. [8] proposed "A vision and Speech Enabled, Customizable, Virtual Assistant for Smart Environment." It is a graphical frontend and a coordinator that leverages on the services to offer to the user a multimodal and involving interconnection with the connected home automation and smart assistant systems.

Revathi et al. [9] has done the work on "Digital Revolution in Speech and Language Processing for Efficient Communication and Sustaining Knowledge Diversity." In this work, they have studied on the digital revolution of automatic speech recognition. In this paper, the user interface which are more suitable for people are discussed. The automatic speech recognition is subset of NLP. The latter is divided into natural language understanding (NLU) and natural language generation (NLG).

3 Proposed Work

In our proposed model, the various existing issues are covered. The disadvantages of the previous work are the user interaction with the application and the availability of the application. In this work, the application is available 24×7 and is easily used by the user through the website.

3.1 Creating GUI Interface for the Application

This project has a graphical user interface to interact with the user and prompts the user for the input via speech. This user interface (UI) acts as a way of interaction between the user and the application. This project contains a user interface useful for

these two types of people: those with physical disabilities and those of with lack of interaction. To create an effective interaction between the application and the user, we created the face of the application simply called as user interface. The HTML5 is used here to create interactive user interface to display the content in the webpage using browser. Webpage or website provides a fast and easy way for interacting with user, and it is supported in every system with Web browser and Internet connectivity. First, create the webpage with basic information and then create the style with tags to the webpage using CSS so that user feels easy to use by looking at the user interface. Add style to the webpage based on your necessity. Generally, the webpage supports buttons, labels, canvas, list box, images, etc. Here, the webpage uses button, images, and microphone. There are almost all types of tags the webpage and a browser support. Each style tag has some set of attributes width, height, bg, fg, etc. For button tags, we have to include "create button" tag which creates the button to be used in the webpage, and the JavaScript is used to create the function call for the button. The called or action to be performed after user clicks the button, and then the backend algorithms or techniques or function used will be executed and outputs the result through the webpage. At last, the script tag in html to take action against each event triggered by the user is created. The interface of the webpage of conversion has one button to trigger when the user speaks and generate the output speech to the user. In this way, the project used the webpage for prompting the user for the input and made it interactive. It is to summarize that in total, for each type of conversion, we used 1 button for the user to speak and stop listening when the user speaks and stops. The website is created to deploy the voice assistant.

3.2 Transforming Speech to Text

In the first transfiguration, i.e., to convert the speech to text, the implementation did here is by using webkitSpeechRecognition API. The webkitSpeechRecognition is a java script speech API. It makes easy to add voice assistant to the webpage. This API gives you complete control and flexibility over your Web browser's speech recognition features. In this project, we start by seeing if the browser supports the Web Speech API by looking for the webkitSpeechRecognition object. If this is the case, we recommend that the user upgrade their browser. (The API is now vendor prefixed because it is still experimental.) Finally, we set some of the characteristics and event handlers for the webkitSpeechRecognition object, which offers the speech interface. To begin, the user clicks the microphone button to input speech to system. We set the speech recognizer's spoken language "lang" to the BCP-47 value that the user has picked using the selection drop-down list, such as "en-US" for English-United States. It defaults to the lang of the HTML document root element and hierarchy if this is not defined. Chrome speech recognition supports a wide range of languages, as well as several right-to-left languages such as he-IL and ar-E. We call recognition after we have set the language. start() is used to turn on the speech recognizer. It calls the onstart event handler when it first starts recording audio and then the onresult

event handler for each new set of results. The onstart event sends the attribute speech to the speech recognition function and the recognizer sends attributes to the onresult event to transform the speech to text. The speech that is taken as input is measured with length and the transcript function is used to transcript from speech to text.

3.3 Transforming Text to Speech

In the case of the second transformation, i.e., converting the text to speech, the implementation is achieved with the speech synthesis module in HTML. The SpeechSynthesis interface of the Web Speech API is the speech service's controller interface; it can be used to receive information about the device's synthesis voices, start and stop speech, and other instructions. The SpeechSynthesis interface of the Web Speech API is the speech service's controller interface; it can be used to receive information about the device's synthesis voices, start and stop speech, and other instructions. The text in the code is generated as speech using speech synthesis. The language used is "en-US" in this application. The GTTs is a Speech Recognition API which is used to convert speech to text or text to speech.

3.4 Knowledge Abstraction

Gathering, manipulating, and augmenting knowledge are the three stages of knowledge abstraction. These phases derive lot for the virtual assistant's material.

Data gathering:
The first stage is to establish a knowledge foundation. This entails locating the course's major concepts and gathering information about them, with the course's core concepts organized in a hierarchical structure. The data is gathered from the user in this application.
Data manipulation:
The data stored in a form of output text to manipulate and categorize the data according to the data gathering.
Data Augmentation:
This process increases the number of natural language processing model training instances available in Dialog flow.

3.5 Response Generation

Contexts serve the function of overlaying the discussion in such a way that only particular intents can be activated when they are present. Because responses are the outputs of intents, they are triggered whenever a user entry matches intent. Using

precise questions, we can group data coming from student interaction with the bot. It is crucial to make a statement about the difficulties of drafting a response. We understand that how students view slide displays differs from how they should interact with a virtual assistant. This highlights the need of content adaptation or the process of identifying the media that best conveys the desired message. For example, the proposed project has added a location of a college by showing it in the Google Maps (Fig. 1).

The flowchart shows the flow of the data through the application. The user uses user interface to interact with the virtual assistant. The input speech is sent to Speech Recognition API. Giving the text as output to the system, the system checks whether the user is still speaking or not. If the user is speaking, it again changes speech to text and processes and gives the matched keyword result as speech output. Else it will terminate.

3.6 Algorithm

Step 1 Start Process.
Step 2 importing libraries and create an object to the classes.
Step 3 Get Audio from the user
Step 4 Convert speech to Text using GTTS.
Step 5 if the text with Keyword match
then return the corresponding result Response
Step 6 convert Text to speech using GTTS.
Step 7 return the Result.
Step 8 End Process

The process starts by opening the website, then the libraries are loaded, then the application get the audio from the user and convert that audio into speech to text using GTTS API. If the text that changes has a keyword that matches the keyword in the application, then the corresponding result will be outputted by the system. Then, the texts of response are converted to speech using GTTS and return the result to the user. This repeats throughout the processes and ends the processes.

4 Results Analysis

The functions are presented to the users through an interactive and attractive user interface, one for each module. The output and end results could be observed as below. Initially, when the application is made to run the website is opened, a UI as shown in Fig. 2 appears.

Click on the user at the right bottom of the page. As the steps in the user interface suggest, the user is prompted to speak, allow microphone and allow popups, the user

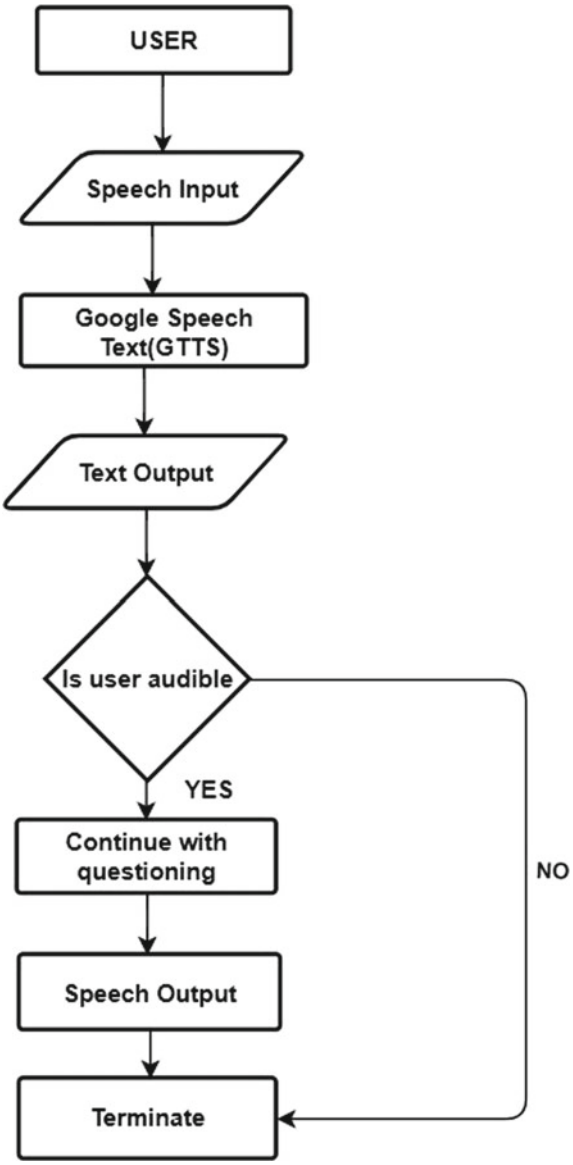

Fig. 1 Flow chart of voice assistant

clicks on the button to speak to the voice assistant, Fig. 3. Once the user speak to the assistant such as open the vr Siddhartha college website, then the college website will be opened as shown in Fig. 4. The user could go on and click on the button to start are stop the voice as shown in Fig. 3.

In Fig. 4, where the website that is commanded by the user to open is opened by virtual assistant is shown.

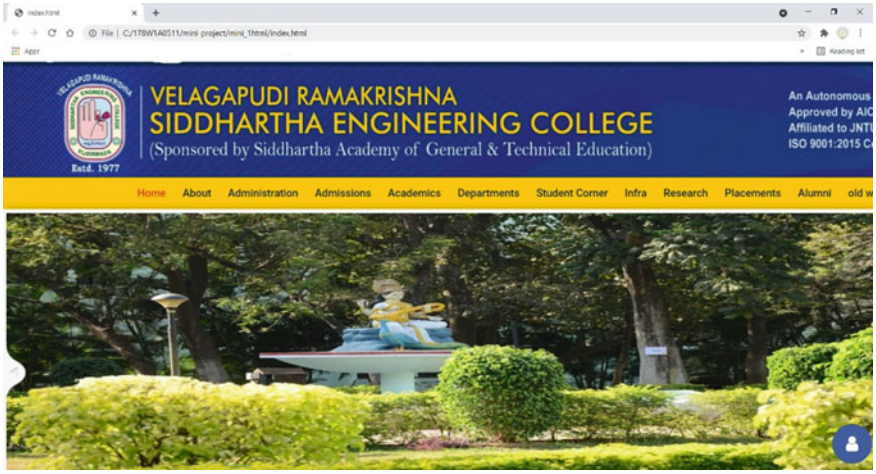

Fig. 2 UI of website to deploy the virtual assistant

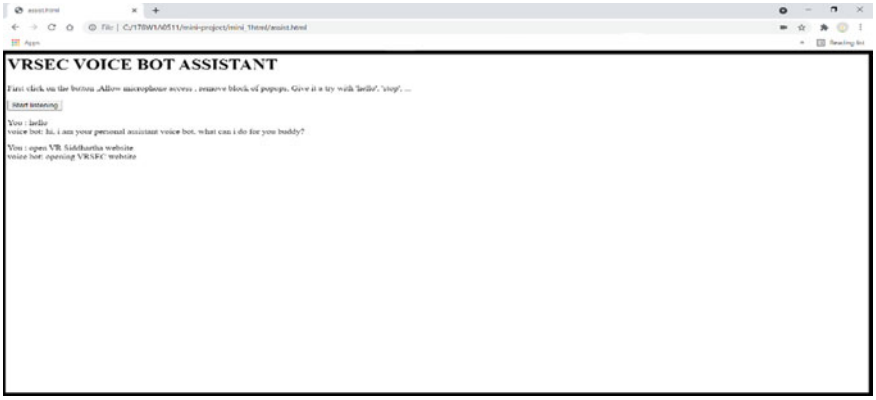

Fig. 3 User commanding the virtual assistant to open another website

So, we tested it out on the opening website, which is included in the code. If the user commands the website that is not included in the code, then the google will get the input of the user and return the results to the user. In Fig. 5, it is the user commanding more works to do by the virtual assistant.

The user has provided orders to the voice assistant to conduct tasks; therefore, the voice assistant opens one by one. We've expanded the use cases to include numerous websites as well. Figure 6 illustrates the commands being executed by opening a page that is not in the code.

We have extended the use cases to locations as well as well. In Fig. 7, it could be seen that a communicated location is also opened.

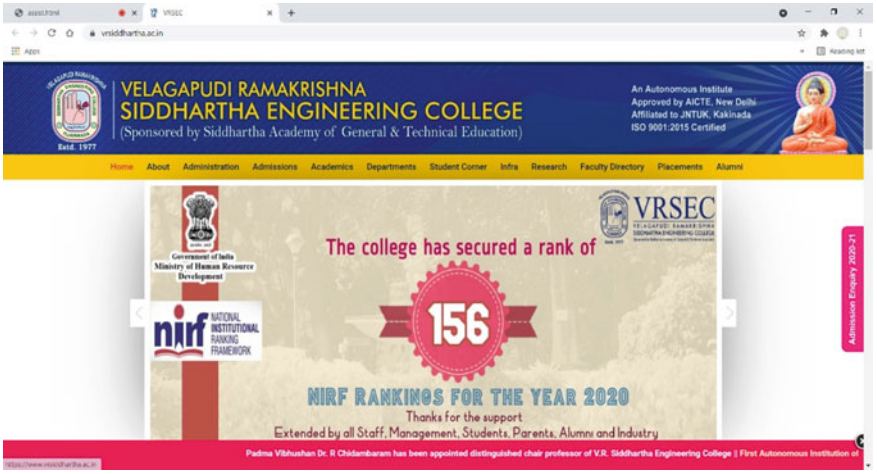

Fig. 4 Output of the website commanded by the user

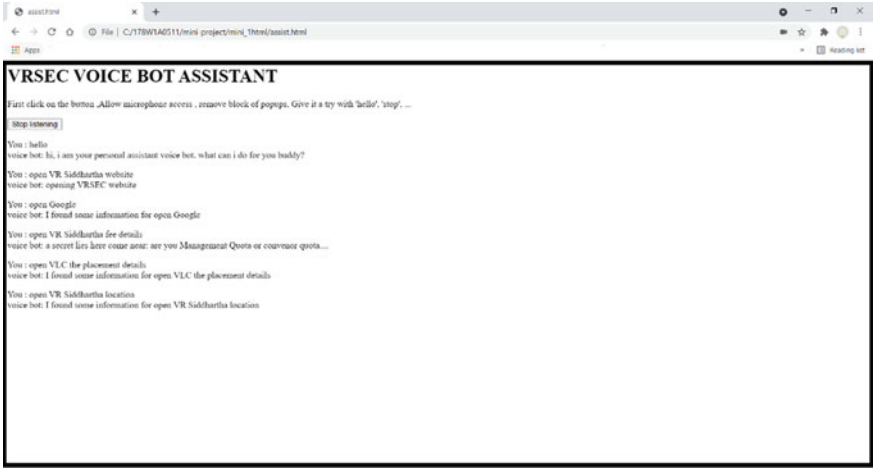

Fig. 5 Recognizing the speech and performing operations commanded by the user

The use cases that we have considered providing the system as an input is a speech as seen in Fig. 5. This is converted to text in the webpage. The resulting output successfully produced the text and speech from the given instructions by user.

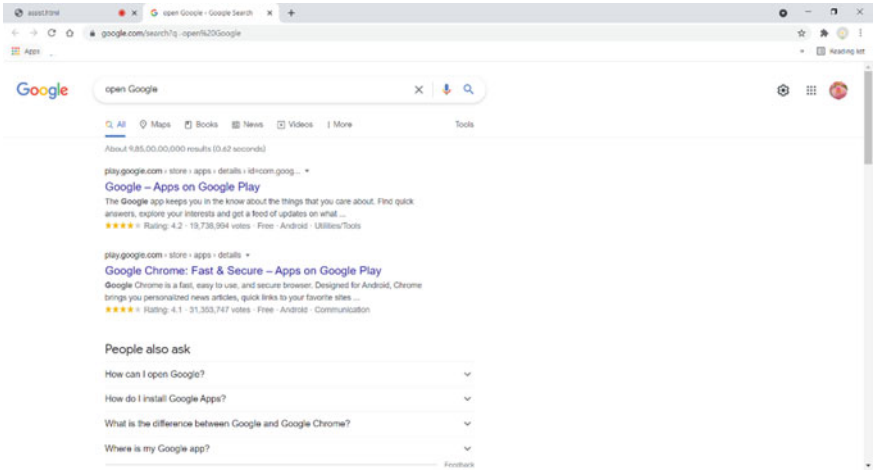

Fig. 6 Opening the page commanded by user

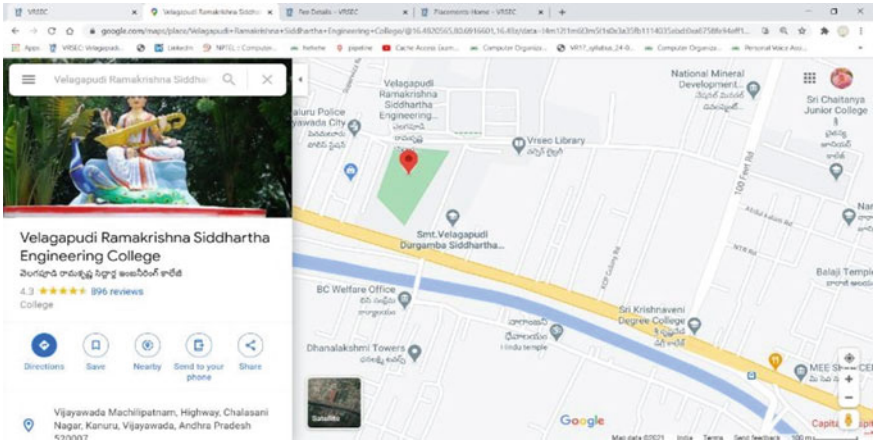

Fig. 7 Extracting the location that the user gave as speech

5 Conclusion

The deployed system is an AI-based college assistant that can help college students with their college-related questions, decreasing confusion and preventing opportunities from being lost. The established system can provide useful information such as college information, location, fee information, and a timetable. The disadvantages of the system are it is not self-learning system, and it is limited to only one college. As a future implementation of the system, we can use neural networks. Data compression and encryption can also be used to save information. Also, a machine

learning module may be added to AI so that trending topics for a specific branch can be recognized by AI based on priority, and users can be notified about the current top priority topics.

References

1. S. Siddesh, A. Ullas, B. Santosh, Aritificial intelligence based voice assistant, in *IEEE International Conference on Security and Sustainability (WorldS4)*, vol 978–1–7281–68234/20 (IEEE, 2020), pp. 593–596
2. S. Gaglio, V.Y. Shetty, S. Das, Artificial intelligent college oriented virtual assistant, in *2019 IEEE International Conference on Smart Computing (SMARTCOMP)*, vol 4, No 2 (2019), pp. 132–137
3. S.V. Prajwal, G. Mamatha, P. Ravi, D. Manoj, S.K. Joisa, Universal semantic web assistant based on sequence to sequence model and natural language understanding, in *International Conference on Advances in Computing and Communication (ICACC)* (IEEE, 2019)
4. D. Pal, C. Arpnikanondt, S. Funilkul, User experience with smart voice assistants: The accent perspective, in *IEEE conference, 10th ICCCNT* (2019)
5. M. Atzeni, M. Atzori, AskCO: A multilanguage and extensible smart virtual assistant, in *IEEE 2nd International Conference on Artificial Intelligence and, Knowledge Engineering* (2019)
6. G. Bohouta, Next-generation of virtual personal assistants (Microsoft Cortana, Apple Siri, Amazon, Alexa and Google Home), in *IEEE 8th Annual Computing and Communication Workshop and Conference* (2018)
7. A. Novikov, I. Itkin, R. Yavorskiy, Development of intelligent virtual assistant for software testing team, in *IEEE, Nineteenth conference on software Quality, Reliability and Security Companion* (2020)
8. G. Iannizzotto, L.L. Bello, A. Nucita, A vision and speech enabled, customizable, virtual assistant for smart environment, in *11th International Conference On Human System Interaction(HSI)* (IEEE, 2018)
9. P.K. Deepthi, P.K. Vasanthi, Digital revolution in speech and language processing for efficient communication and sustaining knowledge diversity. CSI Commun. **41**(7), 6–9 (2020). ISSN 0970–647X
10. A.R. Revathi, P.K. Deepthi, P.K. Vasanthi, Digital revolution in speech and language processing for efficient communication and sustaining knowledge diversity. CSI Commun. **41**(7), 6–9 (2020). ISSN 0970–647X
11. D.S. Zwakman, D. Pal, C. Arpnikanondt, *Usability Evaluation of Artificial Intelligence-Based Voice Assistants: The Case of Amazon Alexa*, (Springers Nature, 2021), pp. 2–16
12. A. Poushneh, Humanizing voice assistant: The impact of voice assistant personality on consumers attitudes and behaviors. J. Retail. Consum. Serv. (2020)
13. G. Daniel, J. Cabot, L. Deruelle, M. Derras, Xatkit: A multimodel low-code chatbot development framework. IEEE Access (2020)
14. D. Carlander-Reuterfelt, A. Carrera, C.A. Iglesias, O. Araque, J.F. Sanchez Rada, S. Munoz, JAICOB: A data science chatbot. IEEE Access (2020)
15. A. Novikov, I. Itkin, R. Yavorskiy, Development of intelligent virtual assistant for software testing team, in *IEEE, 19th conference on software Quality, Reliability and Security Companion* (2020)
16. S. Manoharan, N. Ponraj, Analysis of complex non-linear environment exploration in speech recognition by hybrid learning technique. J. Innovative Image Process. (JIIP) **2**(04), 202–209 (2020)