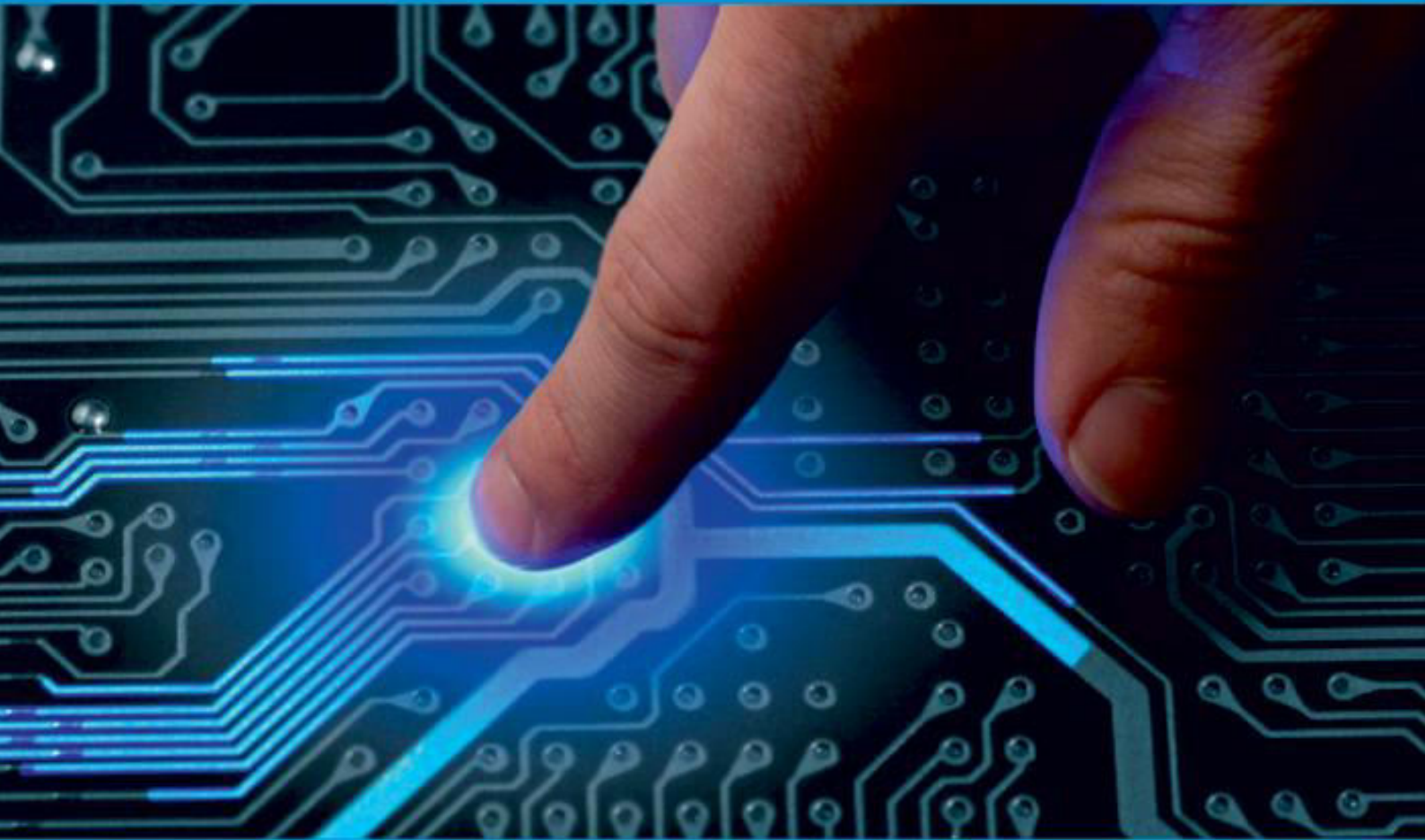




IJIRCCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 12, Issue 3, March 2024

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.379



9940 572 462



6381 907 438



ijircce@gmail.com



www.ijircce.com

Voice Assistance AI

Vrushali. R. Sonar, Pratik. S. Kore, Vaishnavi. I. Deore, Sarthak. V. Divate, Varad. R. Bahirat

Lecturer, Dept. of Computer Engineering, AISSMS Polytechnic, Pune, Maharashtra, India

Student, Dept. of Computer Engineering, AISSMS Polytechnic, Pune, Maharashtra, India

ABSTRACT: Every procedure is now digitalized. This voice-assistance device serves as a basic medical prescriber, daily calendar reminder, note writer, calculator, and search engine. This project accepts voice input and outputs text through speech, as well as displaying it on the screen. Our main goal with our voice assistance is to make person smarter and to give rapid and measured results. The voice assistance system receives voice input via our microphone (Bluetooth or wired) and converts it to computer-comprehensible English. It then provides the user with the required solutions and replies. This assistance connects with the internet in order to provide answers to the user's inquiries. Genuine Language Processing algorithms enable computers to engage in conversation using a variety of kinds of natural human language. Nowadays, python is the most widely used language for artificial intelligence due to its plethora of AI packages; so, in this study, the speech recognition module (python) is critical for the conversion of voice to text.

KEYWORDS: Artificial Intelligence, Python, Voice Assistant, Natural Language Processing

I. INTRODUCTION

Natural language processing and human-machine interaction are the trending topics in technology right now. One of the most explored and popular directions of interaction was based on the machine's interpretation of real human language. No longer does a human learn to speak with a machine; rather, a computer learns to interact with a human by observing his actions, habits, and behaviour and attempting to become his customized assistant. Virtual assistants are computer applications that aid you with daily duties such as displaying weather data, preparing remainders, and building grocery lists, among others. They are capable of receiving commands via text (online chat bots) or voice.

Intelligent assistants that use voice commands require an inciting phrase or wake word before the listener is activated. So many virtual assistants are available to us nowadays, such as Siri, Alexa, and Cortana, Amazon, and Microsoft, respectively. This system was created with desktops in mind. User productivity is increased by personal assistant software, which takes care of the user's everyday duties and feeds them data from an internet source.

A virtual assistant with the ability to make intelligent judgments about normal user behaviours may be built using publicly available data and information, which is why this project was begun in the first place. The creation of the intelligent assistant is affected by the methodologies and approaches used by each company's developer. While a few assistants are able to execute a wide range of activities flawlessly, others are more limited in scope but are nonetheless capable of completing them precisely and to the user's satisfaction. No matter how hard you try, you'll never be able to find a universal helper who can do everything. The attributes of an assistant are determined solely by the developer's focus on a certain area.

The fact that all systems employ machine learning methods and rely on large quantities of data acquired from a variety of sources and then trained on them, the source of this data plays an important role. The assistant's personality is molded by the quantity of data it has access to from various sources. Building such systems, despite the variety in methods to learning as well as algorithm and technique, has remained largely unchanged over time. A human-to-human communication system based on a person's natural language. These include voice activation, speech recognition, Teach-To-Speech, voice biometrics, conversation management, natural language comprehension, and named entity recognition.

II. RELATED WORK

The investigation conducted by Lehlohonolo Mohasi et al. aimed to evaluate the efficacy and functionality of the "Text-to-speech technique in human-computer interaction". This particular system was implemented by employing approaches centered around text-to-speech and human-computer interaction. Geetha v et al. noticed the development of

a voice-enabled personal assistant for PC using the Python programming language. In order to achieve this, they employed various technologies such as text-to-speech, speech recognition, conversational interfaces, and chat bots. These components collectively contributed to the creation of an advanced and personalized assistant that could engage in meaningful conversations.

The study conducted by Farzaneh nasirian et al., titled "AI-Based Voice Assistant Systems: Evaluating from the Interaction and Trust Perspectives", focused on evaluating the quality of interaction, trust, and technological adoption in the context of AI-based voice assistant systems. The authors thoroughly examined the extent to which users could effectively interact with the voice assistant and the level of trust they placed in its capabilities. Additionally, they investigated the users' attitudes towards adopting this technology in their daily lives.

Tae-kook kim et al. conducted a short research study on a voice control system based on Artificial Intelligence assistance. In their investigation, they utilized various technologies such as IFTTT (If This Then That), Raspberry Pi, and Google Assistant to develop a voice control system that integrated AI capabilities. This research aimed to explore the potential of AI assistance in enhancing the functionality and usability of voice control systems.

Lanu Xu et al. conducted a research study titled "CHA: A Caching Framework for Home-based Voice Assistant Systems" in which they developed a caching framework specifically designed for home-based voice assistant systems. Through the utilization of caching and edge computing techniques, the authors aimed to improve the performance and efficiency of voice assistant systems in a home environment. Overall, these studies have made significant contributions to the field of voice assistant systems by exploring various approaches and technologies to enhance their functionality and usability.

III. METHODOLOGY

The proposed system is designed with the capability to launch applications directly by using voice commands and also it has the ability to send E-Mails and WhatsApp messages effortlessly. The functionality of this system includes the ability to read documents as well as access and modify files within the system. The Python programming language is hardly ever utilized for the implementation of the speech recognition module. To enhance the system's functionality, several important modules are incorporated.

One of the significant modules integrated into the proposed system is the Pywhatkit module. Python offers a wide range of built-in libraries that simplify our work, and the pywhatkit module is one such example. This module provides various features, including the ability to send messages through WhatsApp, play YouTube videos based on a provided link, conduct Google searches, and obtain information about specific subjects. In addition, the pywhatkit module can also be used to create handwritten text graphics. With its versatility, this module enhances the overall capabilities of the system. Let's delve deeper into the features provided by the pywhatkit module. Firstly, it enables the sending of WhatsApp messages seamlessly. By leveraging the WhatsApp web in the browser, the pywhatkit module allows users to send messages to any registered mobile number. As long as the browser remains active, this feature can be utilized effortlessly. Secondly, the module facilitates playing YouTube videos by utilizing the pywhatkit playonyt() function. When this function is invoked, the default browser opens, and the specified video is played. Furthermore, if a topic is provided as an option, the module can even play a random video related to that topic. Additionally, users can also provide the URL of a specific video they wish to watch.

Another noteworthy feature of the pywhatkit module is the ability to conduct Google searches. By executing a simple command in the terminal, users can command the code to open the browser and perform a search based on the specified topic. This characteristic boosts the functionality of the system by offering users swift access to pertinent information from the internet. Additionally, the module allows users to obtain a concise overview of a specific subject. It is worth observing that the number of lines exhibited on the screen can be restricted for improved legibility. However, it is important to ensure that the search is bounded to subjects available on Wikipedia.

Moving on to another module incorporated in the system, the SMTPLIB module plays a crucial role in enabling the sending and receiving of emails. The Simple Mail Transfer Protocol (SMTP) it helps to do transmission of emails between mail servers. With the smtpplib module, any internet system that has an SMTP or ESMTP listener daemon can receive mail. This module defines an SMTP client session object, which facilitates the smooth transfer of emails.

Furthermore, the functools module is utilized in the proposed system to provide support for higher-order functions. These higher-order functions are designed to work on other functions, makes it easier to interact with and extend them without the need for complete rewriting. The functools module consists of two classes: partial and partial method. The partial class is particularly useful for creating partial functions, which are functions that only apply to a specific set of arguments. By utilizing the functools library's "partial" functionality, developers can generate these functions in Python. It is important to note that these properties are not formed automatically and require the programmer's intervention to create them. Partial() objects possess three read-only characteristics, which further enhance the flexibility and functionality of the system.

The proposed system incorporates various modules to enhance its capabilities. The pywhatkit module, with its features such as sending WhatsApp messages, playing YouTube videos, conducting Google searches, and obtaining information, greatly enriches the system's functionality. Additionally, the SMTPLIB module enables smooth email communication, while the functools module provides support for higher-order functions through its partial class. These modules collectively help at the overall effectiveness and efficiency of the proposed system. A partial class refers to a class that only applies to a specific set of arguments, thereby making it a specialized function. In Python, the functools library provides the "partial" function, which can be utilized to generate partial classes. Neither like regular classes, these partial classes are not automatically formed, and it is the responsibility of the programmer to create them. The partial() objects have three read-only characteristics, which distinguish them from other classes.

Tkinter programming is a prominent aspect of Python's graphical user interface (GUI) development. Tkinter serves as the GUI library for Python, enabling developers to easily create visually appealing and interactive user interfaces. This is made possible by the integration of Python with Tkinter. Tkinter utilizes the object-oriented interface provided by the Tk GUI toolkit. Tkinter simplifies the process of creating a graphical user interface for your program. By following a set of instructions one can quickly develop a GUI application. The starting step is to import the Tkinter module, which grants access to all the necessary functionalities. Once imported, you can proceed to create the primary window for your GUI program. This window serves as the main interface for the user. To enhance the functionality and interactivity of your GUI application, you can incorporate one or more widgets provided by Tkinter. Widgets like buttons, text boxes, labels, and various other elements that allow the user to interact with the program. By strategically placing these widgets within the GUI interface, you can create a user-friendly experience. Once you have designed your GUI program, it is essential to activate the main event loop. This loop ensures that the program can respond to user input efficiently. When a user interacts with a widget or performs an action, the main event loop detects and handles the event accordingly. It enables the program to provide real-time feedback and maintain a smooth user experience. Partial classes in Python are specialized functions that only apply to specific arguments, and Tkinter programming simplifies the process of developing graphical user interfaces. By following a set of instructions, developers can import the Tkinter module, create a primary window, incorporate various widgets, and activate the main event loop to ensure responsiveness to user input

IV. SYSTEM DESIGN & ARCHITECTURE

Architecture of the proposed system: Our proposed system for Voice Assistant A.I is an intricate amalgamation of machine learning and artificial intelligence. It consists of a user-friendly interface specifically tailored for healthcare related professionals and caregivers, facilitating the process of diagnosing patients. The system will employ machine learning models that have been trained on meticulously pre-processed data in order to generate precise predictions regarding autism, thereby providing valuable insights to aid in early intervention and support. The system's notable characteristics include measures to ensure data privacy, integrity, confidentiality, ethical considerations, and the ability to interpret results, all of which contribute to maintaining transparency in the decision-making process. The system has been designed to be scalable, accessible, and adaptable, with great potential for more advancement. It is important to note that this system is not merely a tool, but rather a particular stride towards creating a society that is well-informed, where autism is identified and managed with outmost efficiency and empathy.

4.1 System Algorithm:

The step which is mandatory in the process is that it involves the analysis of the microphone commands made by the user. This task encompasses a wide range of activities, such as retrieving data and managing the contents stored within a computer. The current study is empirical qualitative research that involves the examination and testing of cases described in the literature mentioned earlier. To gain a better understanding of the Voice Assistant depicted in Figure 1, tests are conducted by programming in accordance with various books and internet resources. The Voice Assistant comprises three main components. Firstly, the assistant accepts the user's voice as input. Secondly, it analyzes the user's input and translates it into the appropriate function and intent. Lastly, the assistant delivers the outcome to the user

through voice. The underlying principle of the voice assistant is illustrated as follows. In order to convert spoken input into written text, voice recognition software is utilized. Subsequently, the central processor employs this text to determine the type of command and then calls upon the appropriate script to carry it out. Nevertheless, there is a multitude of factors beyond these basic steps. Despite dedicating considerable time to the process, the program may still struggle to comprehend the spoken words. Background noise can significantly disrupt a speech-recognition system. As an example to recognize dog's bark or the sound of a helicopter passing overhead and the user's voice is beyond the system's capabilities. To overcome this challenge, engineers must instruct the device to filter out such background noises by collecting data and programming it accordingly and an important consideration is that the individuals naturally modify the pitch of their voice to adapt to a noisy environment. Speech recognition algorithms possess the ability to detect these fluctuations in pitch.

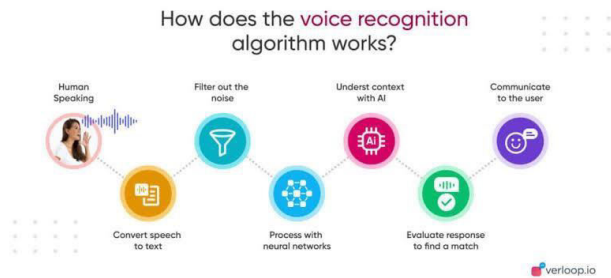


Fig 1 algorithm

4.2 System Flowchart:

The graphical user interface which is known as GUI is displayed by the system, which then enters the listening mode. Voice of the user is recognized as an input by the system and converted to text through the use of the speech recognition module. Subsequently, the system verifies whether the command is linked to any specific action or task. If there is a task associated with the command, the system proceeds to execute it accordingly. For instance, if the command entails searching for information on Wikipedia, the output of this search is converted from text to speech using the Python library. The system then vocalizes the resulting information. Conversely, if the command does not correspond to any task, the system responds with a speech message stating, "Sorry, please try again." In such cases, the system returns to the listening mode and resumes the process.

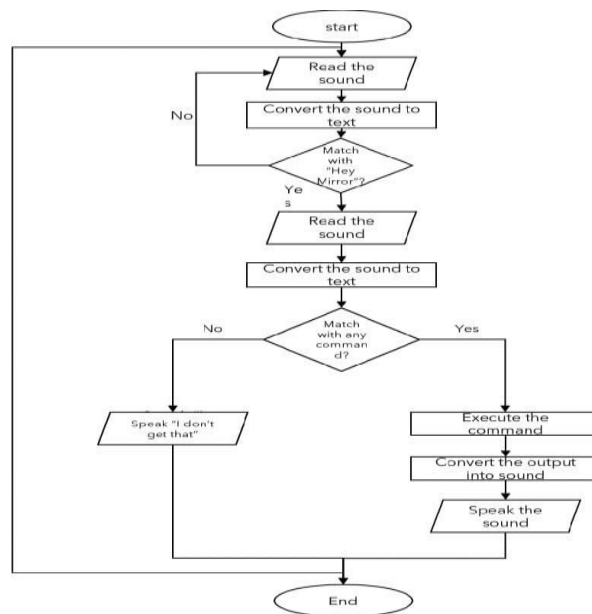


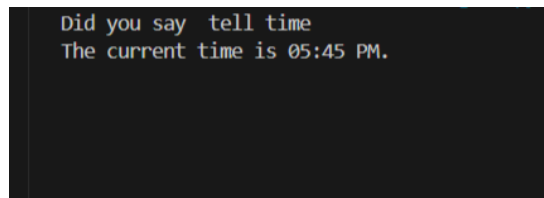
Fig 2 Flowchart

V. IMPLEMENTATION

- 5.1 The speech recognition module:** It utilizes Google's online voice recognition technology to convert spoken input into text. When a user speaks into the microphone at an information center, the audio is stored and sent to Google Cloud for voice recognition. The transcribed text is then received and processed by the central processor.
- 5.2 Python Backend:** Upon receiving the output from the speech recognition module, the python backend determines whether the command or voice output corresponds to an API Call and Context Extraction by comparing it to the command's context. Later, the python code is executed on the user's computer to generate the desired outcome.
- 5.3 Application Programming Interface (API):** It refers to a software mediator that enables communication between two programs. An API acts as a messenger, conveying your request to the service provider and returning the result to you.
- 5.4 Context extraction (CE):** It hopes to extract structured information from unstructured or semi-structured machine-readable materials. It involves utilizing natural language processing (NLP) to process texts in human languages. Recent multimedia document processing operations, such as automated annotation and content extraction from photos, audio, and video, provide test results for context extraction.
- 5.5 Text-to-Speech (TTS):**
It represents a computer's capacity to articulate textual content. Through a TTS Engine, printed text is transformed into phonetic waveforms that can be played back as audio. Third-party publishers offer TTS engines in various languages, dialects, and specialized vocabularies.

VI. RESULT

1. Asking for time: When we speak or ask for time it will display the current time. As shown in fig 3



. fig 3 Displaying Time

2. Redirecting to website: Redirecting to website when we give voice command for it. For example we gave command to open YouTube and it opened the YouTube Website as shown in fig 4

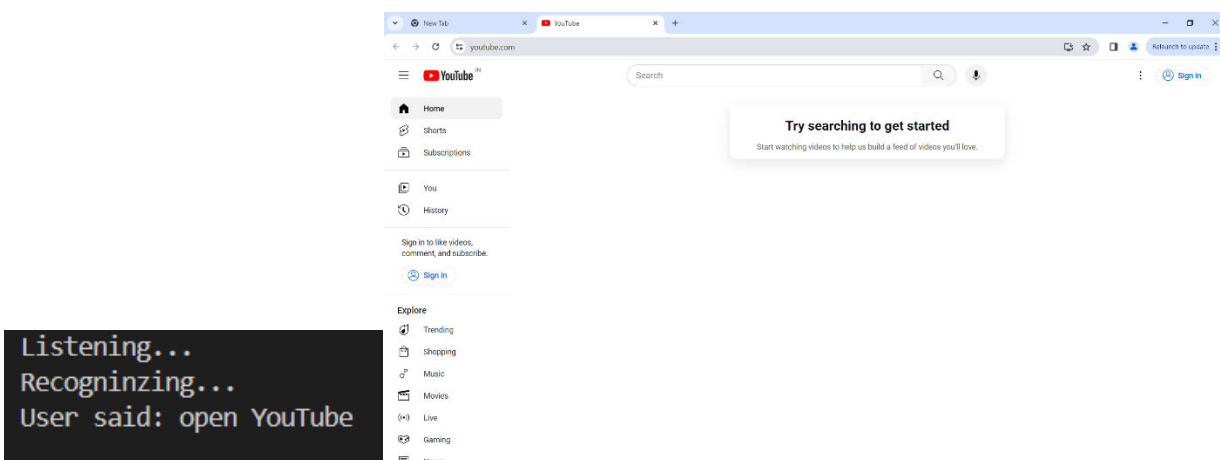


Fig 4 Redirecting to YouTube Website

3. Voice Calculations: It can do mathematical calculations using voice command. We asked for square root of number and it displays the square root of the number in fig 5

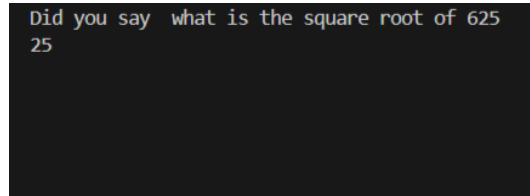
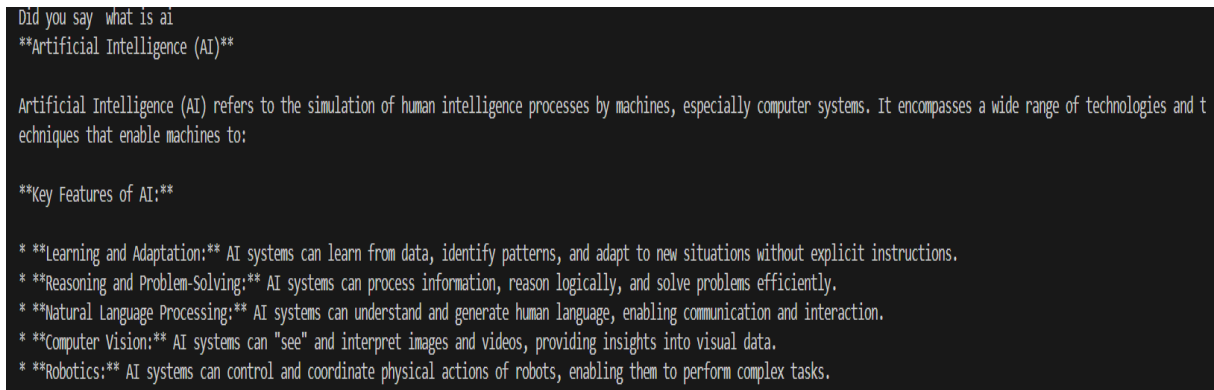


Fig 5 Calculation

4. Information searching: It will display Information or can give the answer to the question as it is asked like we are interacting to human, taking as voice command or either in written. For example we asked it a question and it displayed the information about it in text and voice both.



VII. CONCLUSION

AI (Artificial Intelligence) is a growing field of study which has greater impact on computer science around the globe, that deals with a machines intelligence and capabilities for acting human like and performing human activities. The utilization of AI technology is gradually less in common sectors such as healthcare, banking, and transportation and many more. The potential of AI to upgrade efficiency, take precision, and decision-making has the capacity to revolutionize various industries. The AI's potential to acquire knowledge and enhance performance is facilitated through the application of machine learning algorithms. These algorithms give robots the ability to process extensive datasets, enabling them to recognize patterns and obtain insights that exceed human capabilities. This advancement has propelled the development of natural language processing, computer vision, and speech recognition and it is imperative to acknowledge that the implementation of AI may result in unemployment and the perpetuation of bias in decision-making processes thus, it is crucial for AI research and development to incorporate ethical considerations. While AI has the great potential to improve our society and quality in life, it must be used in a responsible manner.

REFERENCES

- [1] Mohasi L, Mashao D. Text-to-Speech Technology in Human-Computer Interaction. In 5th Conference on Human Computer Interaction in Southern Africa, South Africa (CHISA 2006, ACM SIGHI) 2006 (pp. 79-84). https://www.academia.edu/5906159/Text_to_speech_technology_in_human_computer_interaction
- [2] Elshafei, Moustafa. (2022). Virtual Personal Assistant (VPA) for Mobile Users. https://www.researchgate.net/publication/228748656_Virtual_Personal_Assistant_VPA_for_Mobile_Users
- [3] T. -K. Kim, "Short Research on Voice Control System Based on Artificial Intelligence Assistant," 2020 International Conference on Electronics, Information, and Communication (ICEIC), 2020, pp. 1-2, doi: 10.1109/ICEIC49074.2020.9051160. <https://ieeexplore.ieee.org/abstract/document/9051160>
- [4] Maedche A, Legner C, Benlian A, Berger B, Gimpel H, Hess T, Hinz O, Morana S, Söllner M. AI-based digital assistants. Business & Information Systems Engineering. 2019 Aug;61(4):535-44. https://www.researchgate.net/publication/333705636_AIBased_



Digital_Assistants_Opportunities_Threats_and_Research_Perspective

[5] Xu L, Iyengar A, Shi W. CHA: A caching framework for home-based voice assistant systems. In 2020 IEEE/ACM Symposium on Edge Computing (SEC) 2020 Nov 12 (pp. 293-306). IEEE.

Publication Since 2012 | ISSN: 2321-9939 | ©IJEDR 2022 Year 2022, Volume 10, Issue 2

IJEDR2202018 International Journal of Engineering Development and Research (www.ijedr.org) 112

<https://weisongshi.org/papers/xu20-CHA.pdf>

[6] Shalini S, Levins T, Robinson EL, Lane K, Park G, Skubic M. Development and comparison of customized voiceassistant

systems for independent living older adults. In International Conference on Human-Computer Interaction 2019 Jul 26 (pp. 464-479). Springer, Cham.

<https://mospace.umsystem.edu/xmlui/bitstream/handle/10355/72281/ShaliniShradhaResearch.pdf?sequence=1>

[7] S. Subhash, P. N. Srivatsa, S. Siddesh, A. Ullas and B. Santhosh, "Artificial Intelligence-based Voice Assistant," 2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4), 2020, pp. 593-596, doi: 10.1109/WorldS450073.2020.9210344.

<https://ieeexplore.ieee.org/document/9210344>

[8] Audrey Desjardins, Afroditi Psarra, and Bonnie A. Whiting. 2021. Voices and Voids: Subverting Voice Assistant Systems through Performative Experiments. In Creativity and Cognition (C&C '21). Association for Computing Machinery, New York, NY, USA, Article 29, 1–10. <https://doi.org/10.1145/3450741.3466807>

[9] Zwakman DS, Pal D, Arpnikanondt C. Usability Evaluation of Artificial Intelligence-Based Voice Assistants: The Case of Amazon Alexa. SN Comput Sci. 2021;2(1) 28. doi:10.1007/s42979-020-00424-4. PMID: 33458698; PMCID: PMC7798382. <https://europepmc.org/backend/ptpmcrender.fcgi?accid=PMC7798382&blobtype=pdf>

[10] Natale S. To believe in Siri: A critical analysis of AI voice assistants.



INNO  **SPACE**
SJIF Scientific Journal Impact Factor
Impact Factor: 8.379



ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

 **9940 572 462**  **6381 907 438**  **ijircce@gmail.com**



www.ijircce.com

Scan to save the contact details