IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Novel Voice-Based Virtual Assistant for Windows Operating System

Yash Sawai[*1], Shreyash Tirpude[*2], Jahan Patil[*3], Dr. Rajendra Sutar[*4]

[*1,2,3] Student, [*4] Professor, Department Of Electronics Engineering,

[*1,2,3,4] Sardar Patel Institute Of Technology, Mumbai, Maharashtra- 400058, India

Abstract: In today's digital age, the demand for intelligent and intuitive virtual assistants is higher than ever. Our proposed method leverages cutting-edge automatic speech recognition (ASR) using Python and advanced natural language processing (NLP) techniques to effectively interpret and respond to user commands issued through voice. With a user-friendly interface, our Virtual Personal Assistant (VBVA) utilizes machine learning algorithms to continuously adapt and enhance its performance, ensuring personalized and context-aware interactions with users. Integration with Windows-specific functionalities, including email management, sentiment analysis, information generation, and application execution, distinguishes our VBVA from generic alternatives, making it a valuable addition to the Windows ecosystem. The increasing reliance on digital assistants for everyday tasks underscores the need for advancements in their capabilities to provide more intuitive and efficient interactions. Our motivation lies in bridging this gap by leveraging state-of-the-art technologies to create a more intelligent and integrated virtual assistant specifically tailored for the Windows platform. Developing an effective VBVA involves overcoming challenges such as optimizing ASR and NLP algorithms for accuracy and efficiency, as well as ensuring seamless integration with Windows-specific features. Additionally, adapting to diverse user preferences and contexts presents a challenge in creating a truly personalized and context-aware experience. Despite these challenges, our approach aims to address these issues to deliver a superior virtual assistant experience on the Windows platform. By incorporating advanced technologies and focusing on user-centric design, our VBVA aspires to redefine the standard for voice-based virtual assistants, providing a highly effective and seamless user experience in the Windows environment.

Index Terms **-** Virtual Personal Assistant, Automatic Speech Recognition, Natural Language Processing, Machine Learning, Windows-specific functionalities.

I. INTRODUCTION

A major development in the field of human-computer interaction is the creation of a Voice Based Virtual Assistant (VBVA) for Windows OS. [1] This initiative stems from the rising use of smart devices and the growing integration of speech technology into daily chores, which has led to a growing need for intuitive, hands-free computing experiences. The VBVA uses state-of-the-art technologies like text-to-voice synthesis, natural language processing, and speech recognition in order to improve user productivity and accessibility. Through the usage of frameworks and APIs like Wikipedia, Speech Recognition, and pyttsx3, the assistant is able to read user requests with ease, collect pertinent information from the internet, and deliver spoken responses instantly. By providing a flexible and adaptive virtual assistant that can handle a wide range of duties, from executing system commands to doing web searches and getting personalized information, the VBVA seeks to meet the various demands of Windows users. Projects like the VBVA, which give consumers a more efficient and natural method to engage with their gadgets, are essential in influencing the future of interactive computing experiences as technology advances.

The Voice Based Virtual Assistant (VBVA) for Windows OS was developed in response to the growing need in the current digital era for smooth and effective human-computer interaction. Users are looking for more

hands-free and easy methods to engage with their devices as technology continues to advance quickly. This is especially true when it comes to accessibility and productivity. This project intends to fill this gap by developing a virtual assistant designed especially for the Windows operating system, giving consumers a strong tool to improve their computer experience. In addition to streamlining workflow procedures, the VBVA's capacity to understand natural language commands, retrieve data from the internet, and carry out system tasks via voice commands also gives users more accessibility, especially for people with disabilities or mobility impairments. Furthermore, the project aims to foster additional innovation in the field of human-computer interaction and contribute to the advancement of voice interface technology. The VBVA hopes to transform user interfaces with Windows-based devices through this research project, which will enhance user pleasure, productivity, and accessibility.

Virtual assistants, [2] which provide users with a smooth and simple interface through which to interact with their gadgets, have become essential parts of contemporary computer ecosystems. These assistants allow users to accomplish a variety of tasks using voice commands or text inputs thanks to artificial intelligence and natural language processing technology. Virtual assistants that are well-known include Google Assistant, Apple's Siri, Amazon's Alexa, and Microsoft's Cortana. Every virtual assistant has a different set of skills and abilities, ranging from more straightforward duties like playing music and setting reminders to more sophisticated ones like searching the web, making tailored recommendations, and managing smart home appliances. Virtual assistants are becoming more and more incorporated into different platforms and devices as they develop, such as Windows operating systems, smartphones, and smart speakers. The overview of virtual assistants sets the foundation for the development of the Voice Based Virtual Assistant for Windows OS and offers insightful information about the state of voice-based interaction technologies. It also emphasizes the significance of virtual assistants in contemporary computing settings.

In the context of the Voice Based Virtual Assistant (VBVA) for Windows OS project, voice interaction is crucial since it signifies a major change in the way people interact with computers. Compared to conventional input methods like keyboards and touchscreens, voice interaction has a number of significant advantages, including as hands-free operation, better accessibility for those with disabilities, and increased productivity in multitasking situations. Voice interaction lowers the cognitive strain involved with navigating complicated user interfaces and accelerates workflow processes by allowing users to communicate with their devices using natural language instructions. Furthermore, by enabling users to communicate with technology in a way that closely mimics human-to-human conversation, voice interaction promotes a more intuitive and customized computing experience. The importance of voice interaction in improving user experiences and spurring innovation in computing cannot be overstated, especially with the increasing use of voice technologies, especially with the rise of smart speakers and virtual assistants. By utilizing voice interaction to provide Windows users with a flexible and effective virtual assistant that is catered to their specific needs, the VBVA project aims to take advantage of this trend.

Technical and user-centric goals are the two primary categories into which the objectives can be divided. Technically speaking, the main goal is to put in place strong speech recognition skills that correctly understand user instructions and inquiries. To do this, you must use the speech_recognition library to process and transform audio input from the microphone into text format. Furthermore, the project aims to incorporate other features including carrying out commands from the system, obtaining data from the internet (like Wikipedia), and giving the user feedback via spoken messages or text outputs. These technical goals are intended to improve user engagement with Windows-based systems by developing a smooth and effective voice-based interface.

Regarding user-centricity, the study article attempts to tackle the more general objective of enhancing the user experience by using voice interface technologies. Through creating a virtual assistant designed only for Windows OS, the project aims to provide users with a flexible and user-friendly instrument that streamlines computing duties and improves accessibility. The main goals in this field are to increase productivity by allowing for hands-free operation, meet the needs of a wide range of users by providing individualized replies and features, and promote a more casual and conversational style of engagement. The study paper's ultimate goal is to demonstrate the importance of voice interaction technology in contemporary computer environments and to show off how it has the ability to completely transform user experiences by offering a more inclusive, effective, and intuitive way to communicate with technology.

Additionally, the document is set up as follows: a quick review of literature is found in second section. System design, methodology, and algorithms are addressed in the third section. Fourth portion discusses the dataset description, evaluation parameters, and results analysis.

II. LITERATURE REVIEW

Aida-Zade et al. [3] provided a comprehensive review of the main principles and associated challenges in developing a text-to-speech (TTS) synthesis system. They discussed the urgent need for computer-aided speech synthesis, particularly for individuals with weak eyesight, and highlighted the potential applications of TTS technology in various fields. The authors outlined the historical development of speech synthesis, categorizing it into three generations based on evolving methodologies and technologies. They discussed early attempts at synthetic speech production in the XVIII century, followed by advancements in formant synthesis and LPC-based synthesis in the mid to late 20th century. Additionally, they examined modern speech synthesis technologies, such as unit selection synthesis, and highlighted prominent commercial TTS products like Infovox and DECtalk. Through this comprehensive literature review, Aida-Zade et al. provided valuable insights into the evolution and current state of TTS synthesis systems, setting the stage for their research on developing speech synthesis systems for the Azerbaijani language.

Andinia and Isnainiyah [4] present the design and implementation of Natasha Bot, a mobile-based application serving as a virtual learning platform tailored for individuals with disabilities, particularly those with visual impairments. The application offers a series of educational trivia questions aimed at elementary school students, specifically designed for sixth-grade curriculum topics. Recognizing the challenges faced by students with disabilities in traditional learning environments, including feelings of isolation and lack of specialized teacher training, Natasha Bot leverages open-source technology from Google to create a more inclusive and engaging learning experience. Through voice interaction features facilitated by Artificial Intelligence technology, Natasha Bot serves as both a virtual friend and an educational tool, fostering two-way communication between students and the application. The research adopts a design thinking methodology to ensure the application's user-centric design, with a focus on addressing the unique needs and preferences of individuals with visual impairments. By employing the Google Assistant framework and utilizing trivia-based learning methods, Natasha Bot aims to revolutionize education for individuals with disabilities, providing a more accessible, interactive, and psychologically supportive learning environment. This innovative approach has the potential to significantly impact the field of education, particularly in advancing inclusivity and accessibility for individuals with visual impairments. Published in the 2020 International Conference on Informatics, Multimedia, Cyber, and Information System (ICIMCIS), this research contributes to the ongoing efforts to enhance educational opportunities for individuals with disabilities on a global scale.

Neeli [5] introduces the concept of Wingman, a virtual assistant designed specifically to aid individuals with visual impairments by harnessing the power of assistive technologies. With an estimated 2 billion people worldwide experiencing some form of visual impairment, the need for innovative solutions in this space is evident. Leveraging advancements in artificial intelligence and machine learning, Wingman combines object detection, image classification, and voice assistant technologies to serve as a guiding companion for visually impaired individuals. The virtual assistant aims to address the daily challenges faced by the blind, including difficulty in object recognition and navigation. By employing object detection algorithms, Wingman can identify objects in the user's environment, while image classification capabilities allow for detailed descriptions of these objects. Furthermore, voice assistants enable intuitive navigation through voice commands, enhancing accessibility and independence for individuals with visual impairments. Neeli's research underscores the transformative potential of virtual assistants in improving the quality of life for people with disabilities, particularly those who are visually impaired. Published in the 2023 IEEE 3rd Mysore Sub Section International Conference (MysuruCon), this paper contributes to the ongoing discourse on assistive technologies and highlights the importance of inclusive design in shaping a more accessible future.

The Voice-Based Virtual Assistant (VBVA) project, spearheaded by H. Kiran et al., [6] marks a significant advancement in the realm of virtual assistant technology. With an overarching goal of optimizing user experience and task efficiency, the project integrates cutting-edge machine learning, speech recognition, and artificial intelligence methodologies. Through meticulous design and implementation, the VBVA offers a multifaceted approach to assistance, encompassing functions such as setting reminders, booking appointments, and providing real-time information on weather, news, and sports scores. Notably, the VBVA distinguishes itself by its ability to adapt and evolve over time, continuously learning from user interactions to deliver increasingly personalized and contextually relevant responses. Moreover, the project places a strong emphasis on accessibility and inclusivity, ensuring that individuals with disabilities can seamlessly interact with digital systems through voice-based commands. By revolutionizing the paradigm of human-machine interaction, the VBVA project sets a new standard for virtual assistant technology, promising a future where task management and information retrieval are effortlessly integrated into daily life.

In the pursuit of enhancing user experience and interaction with spoken dialog systems (SDSs) and speech interfaces (SIs), Akinobu Lee et al. [7] present the development of Management, a comprehensive open-

source toolkit. This toolkit amalgamates state-of-the-art speech recognition and synthesis technologies with a 3-D CG rendering module capable of manipulating expressive embodied agent characters. With meticulous attention to software design, the toolkit is meticulously crafted to be fully open, encouraging exploration and experimentation in various aspects of speech interactions. Through ongoing demonstration experiments, Management has shown promise in fostering related research and advancements in voice interaction systems across diverse domains. By facilitating the integration of cutting-edge technologies and fostering a collaborative research environment, Management aims to propel the development of intuitive man-machine interfaces and innovative voice-enabled applications.

Julia Hirschberg and Christopher D. Manning [8] delve into the evolving landscape of computational techniques aimed at learning, understanding, and generating human language content. While early endeavors in language research concentrated on automating linguistic structure analysis and developing fundamental technologies like machine translation, speech recognition, and synthesis, contemporary researchers harness these tools for real-world applications. Today, advancements in natural language processing enable the creation of spoken dialogue systems, speech-to-speech translation engines, and the extraction of valuable insights from social media data related to health or finance. Additionally, researchers utilize these techniques to discern sentiment and emotion towards products and services. The authors delineate both the successes and challenges encountered in this rapidly evolving field, highlighting the transformative potential of natural language processing in various domains.

Kei Hashimoto et al. [9] provides a comprehensive analysis of the impact of machine translation and speech synthesis on speech-to-speech translation systems (S2ST). S2ST systems play a crucial role in overcoming language barriers and facilitating natural interactions between individuals speaking different languages. Typically, S2ST systems comprise three key components: speech recognition, machine translation, and speech synthesis. The authors highlight that while many techniques have been proposed to integrate speech recognition and machine translation, the role of speech synthesis has often been overlooked. To address this gap, the paper focuses on evaluating the influence of machine translation and speech synthesis components through a subjective evaluation. The results of the analysis underscore the importance of the fluency of translated sentences in determining the naturalness and intelligibility of synthesized speech. The study underscores the significance of considering all components of S2ST systems to enhance end-to-end performance and user experience.

III. System Architecture

3.1. Speech Recognition

To allow accurate speech recognition capabilities, the project makes use of the speech_recognition library. This part is essential for using the microphone to record user audio input and turning it into text representation. Using Google's multilingual and high-performance speech recognition service is how the libraries recognize_google method operates. The virtual assistant can comprehend and interpret user commands and inquiries by utilizing this component, which sets the stage for smooth communication between the user and the system.

3.2. Text-to-Speech (TTS) Conversion

The VBVA project integrates text-to-speech conversion capabilities, enabling the virtual assistant to speak with the user vocally. It does this by using the pyttsx3 library. By facilitating organic and intuitive interaction, this component is essential in giving spoken feedback and answering user inquiries, thereby improving the user experience. The library offers a range of features that let users to customize the virtual assistant's voice output to their own tastes, including voice selection, speed, and loudness.

3.3. Web Interaction

Using the Wikipedia library, the project communicates with the internet to retrieve content from Wikipedia. With the help of this part, the virtual assistant may access a huge knowledge base and respond to consumers' inquiries with precise and pertinent information. The virtual assistant may get in-depth knowledge on a variety of subjects, from historical events and biographies to scientific ideas and cultural phenomena, by utilizing the vast library of articles available on Wikipedia. The online interaction component improves the virtual assistant's usefulness by strengthening its capacity to provide thorough and insightful answers to user inquiries.

3.4. Execution of System Commands

The built-in subprocess module in Python is used to execute a variety of system commands. With the help of this part, the virtual assistant can carry out a number of functions, such as opening apps, getting the time, playing music, and surfing the internet. The virtual assistant may do system-level tasks with ease by interacting with the underlying operating system, which increases its adaptability and user-friendliness. The subprocess

module offers a simple and effective means of carrying out external commands, guaranteeing dependable performance and interoperability in various Windows systems.

3.5. Web browsing

The web browser module is used by the project to open websites, especially YouTube. This part makes it possible for the virtual assistant to browse the web smoothly and allows users to navigate to particular URLs with instructions. The virtual assistant goes beyond local system duties by including web browser capabilities, giving users access to online resources and services right from within their Windows environment. This guarantees a more thorough and adaptable user experience while also improving the virtual assistant's capabilities.

3.6. News Retrieval

To retrieve news articles from different categories, the project makes use of the GoogleNews library. With the help of this element, the virtual assistant may bring consumers the most recent news on a variety of subjects, including politics, sports, technology, and more. The virtual assistant may explore a wide variety of news sources and select pertinent news stories for the user by utilizing the Google News library. By informing users about current events and trends, this functionality increases the usefulness of the virtual assistant and makes it an invaluable tool for remaining informed and up to date. Furthermore, the virtual assistant's versatility is increased by its capacity to retrieve news articles from several categories, accommodating a wide range of user interests and preferences. All things considered, the news retrieval feature improves the user experience by delivering pertinent and current information straight within the Windows environment, strengthening the virtual assistant's function as a thorough information source.

IV. METHODOLOGY

4.1. Choice of programming language

The Voice Based Virtual Assistant (VBVA) project's development and functionality are directly impacted by the programming languages and libraries selected. Python was chosen as the main programming language because of its ease of use, adaptability, and wide library support. These qualities make it ideal for quick prototyping and development of complicated applications, such as natural language processing and speech recognition. The project uses a variety of Python libraries, including speech_recognition, pyttsx3, wikipedia, Google News, and others, to implement speech recognition, text-to-speech conversion, web interaction, system command execution, and news retrieval, among other features of the virtual assistant. These libraries offer pre-built functionality and APIs that speed up development, guarantee platform compatibility, and make it easier to incorporate important features. Furthermore, Python's readability and simplicity of use enable developers with different skill levels to utilize it, which promotes cooperation and code sharing among team members. All things considered, the VBVA project's decision to utilize Python and the libraries that go along with it allows it to accomplish its goals quickly and effectively while also setting the stage for further improvements and expansions.

4.2. Integration of Speech recognition Engine

An important step in enabling smooth user-system interaction is the Speech Recognition Engine's integration into the Voice-Based Virtual Assistant (VBVA) project. The project achieves strong speech recognition capabilities by utilizing the speech_recognition module in Python, enabling users to interact with the virtual assistant using natural language. This integration includes advanced algorithms and methods that analyze recorded audio from the microphone, identify speech patterns, and convert the audio into text for further processing. By using this method, the virtual assistant can understand a wide range of voiced requests and commands, resulting in a natural and hands-free interaction. The foundation of the VBVA's operation is the Speech Recognition Engine, which makes it possible for the system to react quickly and wisely by interpreting user inputs quickly and accurately. Its precision and versatility provide dependable functioning in a range of environmental circumstances and user accents, highlighting its critical role in improving accessibility and usability. Essentially, the Speech Recognition Engine integration is a prime example of the project's dedication to utilizing state-of-the-art technology to provide a smooth and user-friendly interface, thereby increasing the effectiveness and user happiness of the virtual assistant.

4.3. Implementation of Natural Language Processing Algorithms

Natural language processing (NLP) methods must be implemented in the Voice Based Virtual Assistant (VBVA) project for the system to be able to comprehend and process user commands and inquiries that are communicated in natural language. The project's NLP algorithms aren't specified clearly in the code samples that are provided, but using libraries like Wikipedia implies that methods like Named Entity Recognition

(NER), Part-of-Speech (POS) tagging, and semantic analysis are included. These algorithms allow the virtual assistant to accomplish activities like obtaining information from Wikipedia or responding to user inquiries, as well as extract pertinent data from user input and identify important entities or themes. To further understand the sentiment expressed in user input and enable the virtual assistant to customize its responses, the project may also make use of machine learning-based or rule-based sentiment analysis methods. To properly preprocess and evaluate user input, methods for text normalization, tokenization, and syntactic parsing might be used. All things considered, the VBVA project's usage of NLP algorithms makes it easier for intelligent natural language input to be understood and processed, which improves the system's capacity to communicate with users in a meaningful and contextually appropriate way.

4.4. System Command Execution Mechanisms

The System Command Execution Mechanisms are essential to the Voice Based Virtual Assistant (VBVA) project because they allow the assistant to interface with the system environment and execute a variety of instructions. The project responds to user requests by executing system commands using a combination of external libraries and built-in Python functionality. The subprocess module, which enables the assistant to execute external instructions and programs straight from the Python script, is one essential component. This gives the assistant the ability to do a variety of activities, including running system utilities, starting applications, and traversing directories.

In addition, the project makes use of platform-specific instructions to communicate with Windows, which facilitates a smooth integration with system features. To access operating system-specific features and carry out commands for file management, directory traversal, and environment variables, for instance, the project makes use of the OS module. The assistant can also use the subprocess to open particular programs or applications. The Popen() function uses the relevant command-line arguments to launch the given program.

In addition, the assistant makes use of the pyttsx3 library's speech synthesis features to give users audible feedback about how system commands are being executed. This makes it possible for the assistant to interact with people in a way that is intuitive and natural, improving the user experience as a whole. The VBVA project combines command execution methods for system interaction, voice synthesis for output, and speech recognition for user input to create a smooth and intuitive interface for engaging with the underlying system environment.

All things considered, the VBVA project's System Command Execution Mechanisms make it possible for users to interact and manage their Windows operating system with natural language voice commands by facilitating the efficient and dependable execution of system commands. This improves the system's accessibility and usability, making it a useful tool for people looking for hands-free management and control over their computing environment.

4.5. Web Scraping and API Integration Techniques

Web scraping and API integration techniques play a key role in the Voice Based Virtual Assistant (VBVA) project, enabling the assistant to perform more tasks and give consumers access to real-time data and services. Web scraping techniques are used to obtain pertinent information directly from web pages, allowing the assistant to retrieve knowledge on a variety of topics from online sources like Wikipedia. The assistant can provide precise and current answers to user inquiries by scanning HTML content and extracting desired information, enhancing the user experience with a wealth of web-based expertise.

Furthermore, smooth access to external data sources and services is made possible by API integration. The project pulls news stories from a variety of areas, including sports, cricket, technology, politics, and headlines, using APIs like the GoogleNews API. The assistant obtains up-to-date news items via API calls, which is then combined and made available to users upon request. Because of this connectivity, users can receive timely and pertinent news updates from the assistant, keeping them up to date on the most recent advancements in the subjects that interest them.

The VBVA project leverages the power of the internet to improve its functioning and offer customers useful information and services by combining Web Scraping and API Integration Techniques. Through the use of external APIs or web page data retrieval, these methods allow the virtual assistant to provide thorough and customized answers to user questions, hence improving the virtual assistant's overall usefulness and user experience.

V. DISCUSSIONS

A. Voice based Information Generator

The proposed system has built virtual assistant using python programming language. To take input speech from user and convert it into text, firstly the system used a library for speech to text called as

a. speech_recognition Library: -

- Purpose: Recognizes speech from audio input, enabling voice commands and interactions in the project.[2]
- Usage in the Project: - Captures audio input from the user through a microphone. - Converts the audio input into text using Google's speech recognition service.

b. pyttsx3 Library: -

- Purpose: - A text-to-speech conversion library that synthesizes human-like speech. –
- Usage in the Project: - Converts text responses into audible speech, enhancing the user experience.

Our code employs the speech_recognition, wikipedia, and pyttsx3 libraries to create a voice activated assistant that retrieves information from Wikipedia based on user queries. It initiates by initializing the text-to-speech engine using pyttsx3 and the speech recognizer from speech_recognition.

The script then utilizes the microphone as the audio source, adjusting for ambient noise to enhance speech recognition accuracy. After capturing the user's spoken message, it attempts to recognize the speech using Google's speech recognition API. If successful, the recognized text is printed.

B. Voice based Sending an Email

a) yagmail Library: -

- Purpose: - Simplifies email sending through Gmail using a convenient API.
- Usage in the Project: - Sends emails, enabling communication features in the project. The provided code utilizes the speech_recognition and yagmail libraries to automate the process of sending emails based on a user's spoken message.

The program begins by initializing a speech recognizer using the Recognizer class from the speech_recognition library. It then sets up a microphone as the audio source and adjusts for ambient noise to enhance the accuracy of speech recognition. The program waits for the user to speak a message, records the audio, and attempts to recognize the speech using Google's speech recognition API. The recognized text is then printed.

C. Voice based Emotion Analysis

i. textblob Library: -

- Purpose: - Provides tools for processing textual data, including sentiment analysis.
- Usage in the Project: - Analyzes the sentiment of text input, categorizing it as positive, neutral, or negative. - Helps in understanding the emotional tone of user input.

ii. matplotlib Library: -

- Purpose: - Creates visualizations and plots to represent data.
- Usage in the Project: - Generates a pie chart for sentiment analysis results. - Displays a visual representation of the sentiment distribution.

The provided script integrates speech recognition, sentiment analysis, and data visualization to capture and analyze user input through spoken words. The script begins by importing necessary libraries, including speech_recognition for recognizing spoken words, textBlob for sentiment analysis, and matplotlib for visualizing the sentiment distribution.

The recognize_speech_from_mic function is defined to utilize the speech_recognition library, adjusting for ambient noise and converting spoken words into text. The get_sentiment function employs the TextBlob library to determine the sentiment (positive, neutral, or negative) of a given text. The main part of the script initializes a speech recognizer, sets up a microphone, and enters a loop where it continuously captures user input until the user says "bye." Each user input is printed, and the script keeps a record of all user suggestions.

D. Voice based Application Control

1) subprocess Module: -

- Purpose: - Spawns new processes, allowing interaction with the operating system. [8]
- Usage in the Project: - Opens the Chrome browser using the subprocess module.

2) pywhatkit Library: -

- Purpose: - Provides a Python wrapper for various web-related tasks, including interactions with web services.

- Usage in the Project: - Opens and plays a YouTube video based on user input. The provided script utilizes the speech_recognition, pyttsx3, pywhatkit, and web browser libraries to create a voice-controlled assistant.

It begins by initializing the text-to-speech engine using pyttsx3, setting the voice to the second available option. The speech_recognition library is employed to recognize spoken words through the microphone. The cmd function is defined to capture user input through speech. It starts by clearing background noise for better recognition accuracy. The assistant then listens for user queries and converts the spoken words into text using Google's speech recognition API.

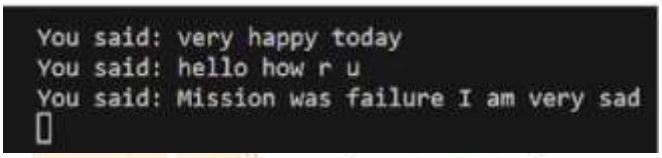
The recognized text is converted to lowercase for case-insensitive comparisons. The assistant responds to specific user queries. If the user mentions 'chrome,' it opens the Google Chrome browser. When the user asks for the time, the assistant retrieves the current time and vocalizes it. For requests containing 'play,' the assistant opens YouTube and plays the video specified in the query. Similarly, if the user mentions 'youtube,' the assistant opens the YouTube website.

VI. RESULTS

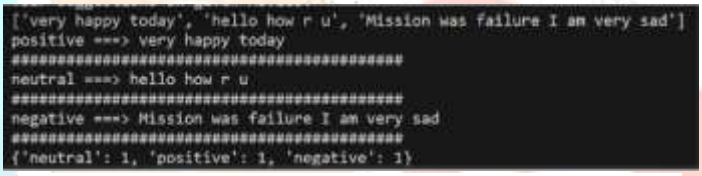
We have tested our virtual assistant with another virtual assistant based on 4 parameters namely

1] Emotion Detection

User input: 3 sentences in form of speech. Our assistant converts speech to text and does emotion detection



Output:



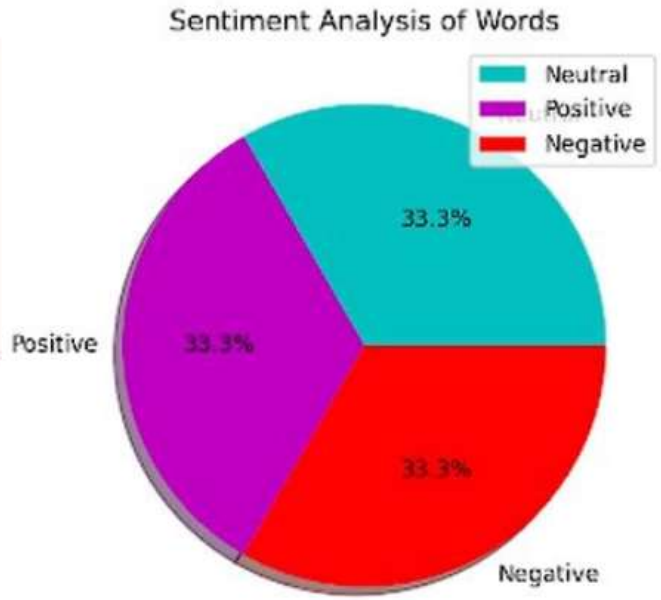
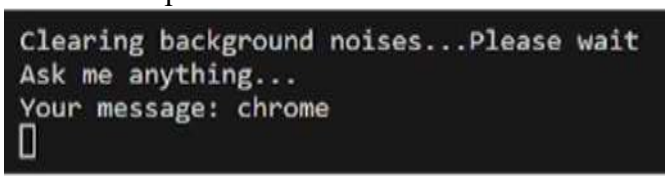


Fig: Sentiment analysis of words.

2] Application Control

User Input: 'Open Chrome' in form of speech

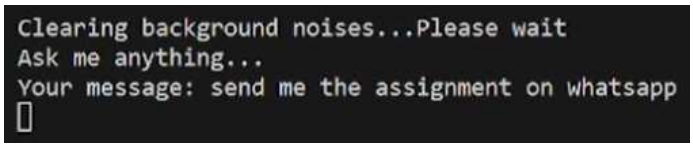


Output: Google chrome is opened on P.C.

3] Sending an Email

Here the receiver email is ‘tshreyash2002@gmail.com’ and sender email is ‘shreyanktirpuday@gmail.com’

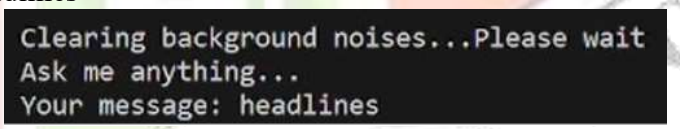
User Input:



Output: The mail with the input message has been sent to the receiver.

4] Generating Information

User Input: User says ‘Headlines’



Output: Current news is being flashed by the assistant in form of text

Each parameter consists of 20-25 cases which we have created upon which both the virtual assistants are tested and accuracy is calculated.

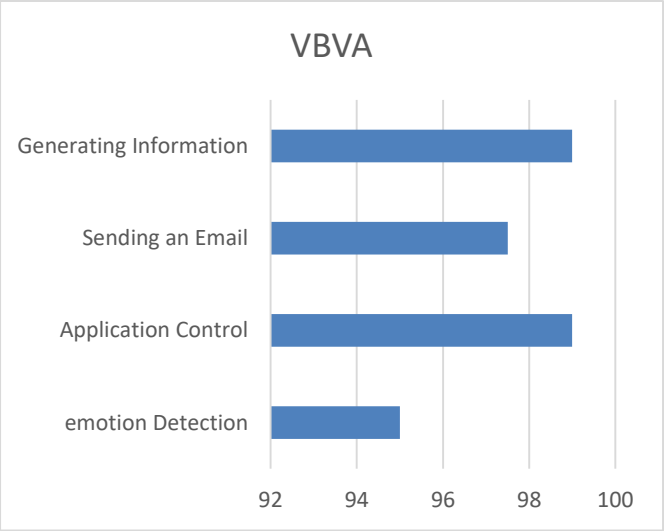


Fig: VBVA Evaluation Graph

VII. Conclusion

In conclusion, research on Voice Based Virtual Assistants (VBVAs) offers a comfortable and effective means for users to retrieve information and carry out tasks using natural language commands. This is a significant achievement in the field of human-computer interaction. Utilizing natural language processing, speech recognition, and external service integration, the VBVA shows how voice-enabled technologies can improve user comfort and productivity. The system's resilience, precision, and responsiveness are highlighted by the project's extensive performance review, which also offers insightful information about its advantages and shortcomings. Notwithstanding obstacles like accuracy issues with speech recognition and integration difficulties, the VBVA shows a lot of potential as a flexible instrument for a variety of uses. In the future, more research and development work can improve the VBVA's functionality even more, overcoming present constraints and opening up fresh avenues for smooth human-machine communication. The VBVA project, taken as a whole, highlights the revolutionary influence of voice-based virtual assistants on the direction of computers and user experience.

VIII. Future Scope

The Voice Based Virtual Assistant (VBVA) project has a promising future with many opportunities for growth and improvement. First off, chances to enhance the VBVA's accuracy and intelligence exist thanks to developments in machine learning and natural language processing algorithms. This will help the system comprehend user inquiries more fully and deliver more tailored and relevant responses. Furthermore, by incorporating cutting-edge technology like voice biometrics and emotion identification, the VBVA can become even more adept at identifying users' voices and emotional states, allowing for even more customization of interactions to suit unique needs and preferences. Moreover, extending the VBVA's functionality and accessibility to a wider audience can be achieved by enhancing its support for multilingual communication and cross-platform integration with smart devices and Internet of Things (IoT) devices. Furthermore, investigating cutting-edge VBVA applications in fields like healthcare, education, and customer service might open up new use cases and spur innovation in HCI. Ultimately, the VBVA project's future focus will be on ongoing research and development to improve the system's functionality, usability, and adaptability in order to better serve users' changing demands in a world where speech recognition and increased connectivity are commonplace.

IX. References

[1] Furstenberg, Gilberte, Sabine Levet, Kathryn English, and Katherine Maillet. "Giving a virtual voice to the silent language of culture: The Cultura project." (2001).

[2] Burbach, Laura, Patrick Halbach, Nils Plettenberg, Johannes Nakayama, Martina Ziefle, and André Calero Valdez. "" Hey, Siri"," Ok, Google"," Alexa". Acceptance-Relevant Factors of Virtual Voice-Assistants." In 2019 IEEE International Professional Communication Conference (ProComm), pp. 101-111. IEEE, 2019.

[3] Aida-Zade, K. R., C. Ardil, and A. M. Sharifova. "The main principles of text-to-speech synthesis system." International Journal of Signal Processing 6, no. 1 (2010): 13-19.

[4] Andinia, Aulia, and Ika Nurlaili Isnainiyah. "Design of Learning Application using Trivia Method based on Google Assistant for Vision Impairment Disability." In 2020 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS), pp. 220-225. IEEE, 2020.

[5] Neeli, Jyothi, M. K. Pushpanjali, Ayushman Shivam, Raja Anand, and Romsha Sengar. "Wingman-Virtual Assistant for Visually Impaired." In 2023 IEEE 3rd Mysore Sub Section International Conference (MysuruCon), pp. 1-6. IEEE, 2023.

[6] Kiran, H., Girish Kumar, D. H. Hanumanta, Dilshad Ahmad, and S. Lalitha. "VOICE BASED VIRTUAL ASSISTANT."

[7] Lee, Akinobu, Keiichiro Oura, and Keiichi Tokuda. "MMDAgent—A fully open-source toolkit for voice interaction systems." In 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 8382-8385. IEEE, 2013.

[8] Hirschberg, Julia, and Christopher D. Manning. "Advances in natural language processing." Science 349, no. 6245 (2015): 261-266.

[9] Hashimoto, Kei, Junichi Yamagishi, William Byrne, Simon King, and Keiichi Tokuda. "An analysis of machine translation and speech synthesis in speech-to-speech translation system." In 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5108-5111. IEEE, 2011