IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 1, June 2023

Sentiment Analysis by Virtual Assistant using Python

Prof. Kavita Patil, Aditi Patil[2], Saloni Patil[3], Sakshi Shinde[4], Shaktiprasad Patra[5]
Professor, Department of Information Technology[1]
Students, Department of Information Technology[2-5]
Zeal College of Engineering and Research, Pune, Maharashtra, India

Abstract: *Artificial intelligence (AI)-driven virtual assistants have been increasingly popular in recent years as a means of enhancing user experience and boosting productivity. The popularity of smart speakers and virtual assistants has raised interest in creating voice-based applications that can accomplish a variety of activities and help consumers. This paper details the use of Python programming to create a voice-based virtual assistant. The assistant understands user requests and responds appropriately using machine learning algorithms and natural language processing (NLP) approaches.Speech recognition systems, commonly referred to as Automatic Speech Recognition, or ASR, are essential for enabling users to interact with virtual assistants.The study also examines Python's interoperability with open-source technology and its potential for creating voice-activated virtual assistants. The article also offers a multi-modal strategy to improve user experience, combining voice- and text-based interactions. The research's findings show how successful the Python-based voice-based virtual assistant is and how it has the potential to simplify user activities and boost productivity.The research also looks at the efficiency of various NLP libraries and the possibilities of Python for constructing voice-activated virtual assistants. Home automation, customer service, and healthcare are just a few of the practical uses of the research.*

Keywords: Artificial Intelligence, voice assistant, Desktop Assistant, Python, Text-to-Speech and speech to text , Virtual Assistant..

I. INTRODUCTION

Artificial intelligence (AI) has changed the way we engage with technology, with virtual assistants becoming more crucial in our daily lives. With the help of virtual assistants, users may easily and intuitively connect with technology to complete tasks like making appointments, setting reminders, and responding to inquiries. The use of voice-based interfaces has increased as a result of the rising popularity of voice-activated assistants like Amazon's Alexa and Google Assistant.Virtual assistants are solely software-based, despite the fact that they are now built into many different devices. Additionally, some assistants, like Alexa, are designed specifically for certain devices. With the recent dramatic developments in technology, we must train our machines utilizing machine learning, deep learning, and neural networks. These days, we can speak to our devices via voice assistant. Every major company today uses voice assistant technology so that clients may ask a machine for help by speaking to it. As a result, with the Voice Assistant, we are moving closer to the day when we can converse with technology. These virtual assistants make it simpler for people to interact with technology, which is especially useful for people who are elderly, physically impaired, blind, or have children.Even visually impaired people, such as the blind, can converse with the device using only their voice.Some of the basic tasks that a voice assistant can help with include the following: - Reading a newspaper, receiving mail updates, searching the web, watching movies or listening to music, setting alarms and reminders, making use of any program, and receiving weather information.

Due to its adaptability and simplicity, Python has grown to be a popular choice for creating virtual assistants. Python is the perfect platform for creating intelligent voice assistants since it offers a variety of frameworks and tools for machine learning and natural language processing (NLP). This study demonstrates the creation of a Python-based voice-based virtual assistant and investigates how effective it is at enhancing user experience and boosting productivity.Machine learning was used to train the model after it had been built using Python modules and tools. The model was additionally enhanced with a few Windows instructions to make sure it would run smoothly on this OS.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 1, June 2023

There are three modes on which our model is based on:
1) Supervised Learning
2) Unsupervised Learning
3) Reinforcement Learning
Depending on the reason for which the user requires assistance. Deep learning and machine learning can be used to achieve these goals. The Voice Assistant will eliminate the need for you to continuously type commands to do specific activities. Once a model is created, it can be used as many times as needed by as many individuals as feasible in the easiest way possible. As a result, we will be able to manage multiple aspects of our surroundings on a single platform with the help of a virtual assistant.

1.1 Aim and Motivation:

The primary aim of this research is to develop and evaluate the effectiveness of a voice-based virtual assistant using Python programming language. The virtual assistant utilizes natural language processing (NLP) techniques and machine learning algorithms to understand user queries and provide appropriate responses. The study compares the effectiveness of popular NLP libraries in implementing the voice-based assistant and explores the potential of Python in developing voice-activated virtual assistants.The research was motivated by the growing desire for intelligent voice assistants that can simplify user chores and increase productivity. Existing virtual assistants frequently lack accuracy and reactivity, causing user irritation. The development of the voice-based assistant using Python and machine learning algorithms offers a possible answer to these challenges.Furthermore, Python has grown in prominence as a programming language for data analysis and artificial intelligence in recent years. Python has a wide selection of tools and modules for natural language processing and machine learning, making it a perfect platform for creating intelligent virtual assistants. The study's goal is to investigate Python's potential in the construction of voice-activated virtual assistants and contribute to the field's advancement.

II. LITERATURE REVIEW

In recent years, there has been substantial study into the development of virtual assistants, with a focus on constructing intelligent agents that can understand and respond to natural language. Natural language processing (NLP) techniques have been widely utilized in the development of virtual assistants, allowing them to comprehend human questions and respond appropriately.
Because of its versatility and ease of use, the Python programming language has grown in favor as a tool for constructing intelligent virtual assistants. Existing virtual assistants such as Amazon's Alexa and Google Assistant have demonstrated the potential of voice-based interfaces in enhancing user experience. Voice-activated assistants offer a more natural and intuitive way for users to interact with technology, reducing the need for complex commands and interfaces. Python provides a range of tools and libraries for implementing voice-based interfaces, including PyAudio, SpeechRecognition, and Pocketsphinx.Using Python and artificial intelligence technologies, a voice assistant that can understand commands and carry out duties assigned by the customer was constructed [1]. The virtual assistant used NLP to interpret user speech or text input into executable commands.Time is saved because the process is rapid. Time is saved because the process is rapid. Because of its modular form, this project is more versatile and straightforward to grasp. The Python programming language's required packages have all been installed, and the code was written using the VS Code Integrated Development Environment (IDE). The data for the various noises came from the environment as well, and the Python version utilized for this project was 3.x.Mobile professionals can use the Virtual Personal Assistant to get an intelligent computer secretarial service [5]. The new service is based on the integration of mobile, internet, and speech recognition technology.The VPA decreases interruptions, improves time management, and serves as a primary hub for all of the user's messages, contacts, schedule, and information sources. The research also proposes a decision-making framework for screening calls and dealing with meeting and appointment requests.The survey offered in this paper will help you understand the differences between IBM Watson and Google Dialogue Flow as Natural Language Understanding Platforms [6].
Development of a portable, largevocabulary voice recognition system that is accurate, fast, and dependable [7].

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 1, June 2023

To achieve real-time performance on modern smart phones, quantized deep neural networks (DNNs) and on-the-fly language model rescoring were used.The literature study on Smart Assistant [4] covers voice recognition algorithms, how SSH can be enabled on Raspberry Pi 4, and facial identification using OpenCV and Raspberry Pi.The voice assistant, which was built with Python, machine learning, and AI algorithms, was designed to help individuals by reacting to their voice commands. The Voice Recognition API is used to convert audio input from the user into an English sentence [3].In conclusion, the literature review emphasizes the importance of natural language processing (NLP) approaches and machine learning in the creation of virtual assistants. Python is a powerful and user-friendly programming language for creating intelligent virtual assistants, particularly ones with voice-based interfaces. In recent years, there has been increased interest in the development of multi-modal virtual assistants, which allow users to move between text and voice-based interactions as needed. The examination of virtual assistants revealed the significance of accuracy and responsiveness in increasing user experience and productivity.

III. PROPOSED SYSTEM ARCHITECTURE

This section summarizes our conclusions after analyzing and comparing our suggested work. This notion was implemented using Python, machine learning, and AI. The user interface, the natural language processing (NLP) engine, and the speech recognition module are the three primary components of the suggested system architecture for the voice-based virtual assistant. These elements are linked and work together to produce a smooth and intuitive user experience.

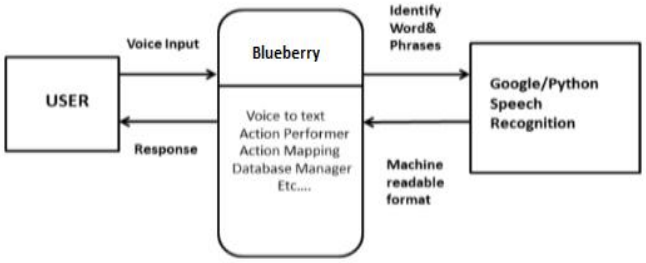


Fig. 1: Data Flow Diagram

User Interface (UI): The user interface is the virtual assistant's front-end component that allows users to engage with the system via voice commands. A mobile app or a web-based interface can be built for the user interface. Users can use the user interface to start a discussion with the virtual assistant, define their requests or questions, and receive responses from the system. To improve the user experience, the user interface might also include graphical components such as photos or movies.

NLP (Natural Language Processing) Engine: The virtual assistant's essential component is the NLP engine, which processes user questions and delivers relevant responses. The NLP engine understands the user's purpose and generates a suitable answer using various NLP approaches such as natural language understanding (NLU) and natural language generation (NLG). The NLP engine can be built with libraries like NLTK, spaCy, or TextBlob.

Speech Recognition Module: The speech recognition module is in charge of transforming the user's spoken words into text format that the NLP engine can process. This task can be performed by the speech recognition module using APIs such as Google Speech Recognition or Microsoft Azure Speech to Text. Libraries like as can be used to implement the speech recognition module.

Sentiment Analysis using TextBlob: The suggested system architecture for the voice-based virtual assistant includes sentiment analysis as an important component, which is based on the TextBlob theory. This enhancement improves the virtual assistant's capacity to perceive and respond to user emotions and sentiments.

1. Speech Input: The system begins by collecting user speech input via a microphone or other voice input device.
2. Automatic Speech Recognition (ASR): The collected speech is converted into text format for further analysis using ASR techniques.
3. TextBlob Sentiment Analysis: The ASR text data is processed by the sentiment analysis module, which employs the TextBlob theory. TextBlob's sentiment analysis interface is straightforward and intuitive, allowing

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 1, June 2023

the system to assess the emotional polarity (positive, negative, or neutral) and subjectivity (opinionated vs. factual) of the user's input.

4. Intent Understanding: The sentiment analysis results are then put into the virtual assistant's intent understanding module. The virtual assistant gains a better understanding of the user's objectives and can modify its responses accordingly by taking into account the user's emotional tone and feeling. For example, if the sentiment analysis reveals a negative sentiment, the virtual assistant can show empathy or offer solutions to any difficulties raised by the user.
5. Response Generation: The virtual assistant creates appropriate responses based on a combined analysis of user intent and sentiment. The responses can be educational, entertaining, or empathic, depending on the user's emotional context.
6. Text-to-Speech (TTS) Synthesis: Using TTS synthesis, the generated response is turned into speech. The synthesized speech is sent to the user via a speaker or other audio output device.

The virtual assistant becomes increasingly adept at understanding and responding to human emotions by incorporating sentiment analysis utilizing the TextBlob theory into the system architecture. This enables the assistant to provide personalized and contextually relevant interactions, hence improving the overall user experience.It is crucial to note that depending on the project needs and available resources, the specific implementation details and algorithms for sentiment analysis utilizing TextBlob may differ. The suggested system architecture provides a generic framework for implementing sentiment analysis into a voice-based virtual assistant, and additional study and experimentation are advised to optimize and fine-tune the sentiment analysis module for various application domains.

Our main goal is to help users with their tasks by using their voice commands. This can be accomplished in two stages: first, taking the audio input from the user and converting it to an English phrase using the Speech Recognition API; second, searching for the task the user wants to perform and then redirecting it to the server using the HTTP Protocol and displaying the result on the web browser.

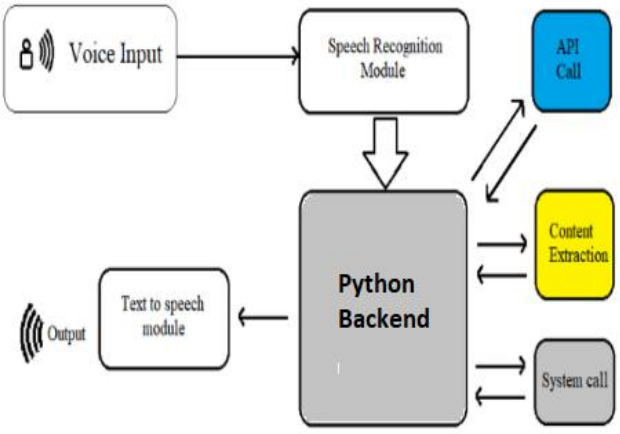


Fig 2: Proposed System Architecture

The proposed system will be able to:

1. Continuously listen for commands, and it will be able to adjust the amount of time it spends doing so based on user preferences.
2. If the system is unable to obtain the information from the user, it will repeatedly ask them to repeat their input.
3. The system can employ male or female voices based on the user's preferences.
4. The current version includes capabilities like as playing music, sending emails and SMS, surfing Wikipedia, opening system-installed programs, and browsing the internet.
5. The system will continue to listen for commands, and the period of that listening can be adjusted based on user requirements.
6. If the system cannot obtain information from it, it will repeatedly ask the user to repeat their input until the desired amount of times.
7. The system can employ male or female voices based on the user's preferences.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 1, June 2023

Some of the main elements in the suggested model:-

1. Weather detection: Include weather detection capability to give customers with the most recent weather information and forecasts.
2. Sentiment analysis: Use sentiment analysis to enable the virtual assistant to recognize and respond properly to the user's emotional condition.
3. Build a reminders module that allows the virtual assistant to set reminders for users based on voice requests.
4. Appointment booking: Create a booking module that allows the virtual assistant to book appointments for users based on voice commands.

These features can be added to the natural language processing (NLP) engine, which will read the user's voice commands and provide appropriate responses based on the input received. To provide more extensive functionality to the virtual assistant, the NLP engine would also interface with additional modules such as weather detection, sentiment analysis, reminders, and booking modules. Overall, the suggested system architecture would provide a flexible and robust framework for constructing a voice-based virtual assistant using Python that can deliver a wide range of features and services to consumers.

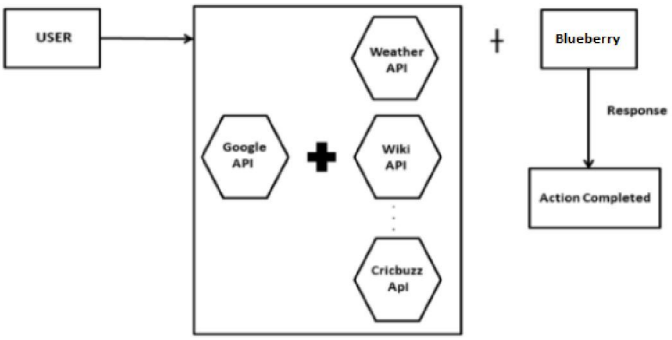


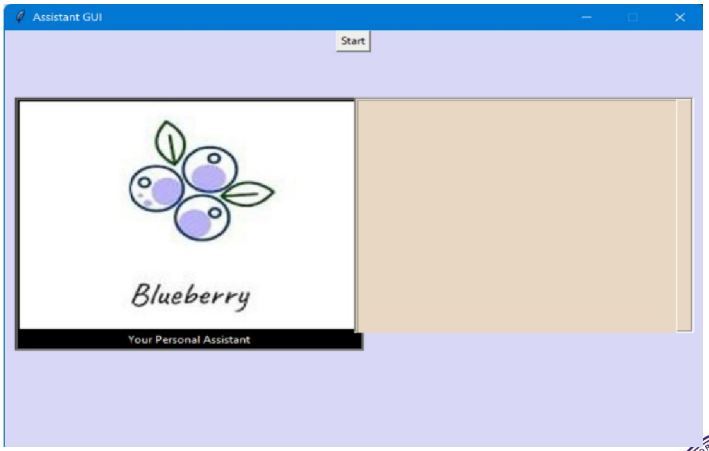
Fig 3: Data Flow Diagram

V. RESULTS

A virtual assistant consumes less time. A virtual assistant is software that recognizes commands and completes tasks provided by the customer. NLP is used in virtual assistants to match user speech or text input with executable commands. You may run your machine, such as a laptop or a PC, on your own with the help of a virtual assistant. It is a quick process, therefore it saves time. A virtual assistant works for you at specific times, so they are constantly available to you and can swiftly adjust to changing needs.Your virtual assistant will be available to you and, if their workload permits, will assist others such as family and colleagues.

VI. IMPLEMENTATION DETAILS

Fig 1. User Interface



IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 1, June 2023

Fig 2: Greetings

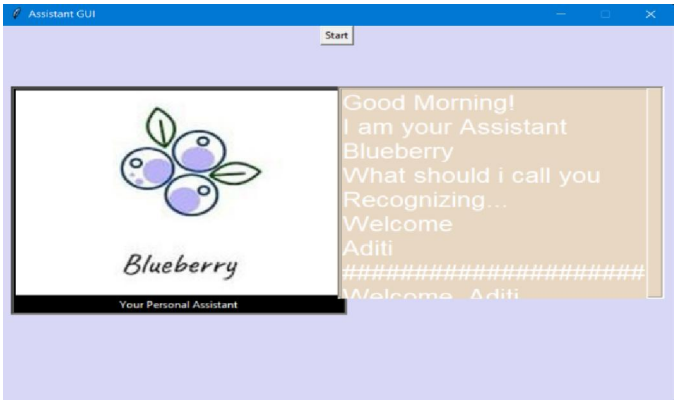
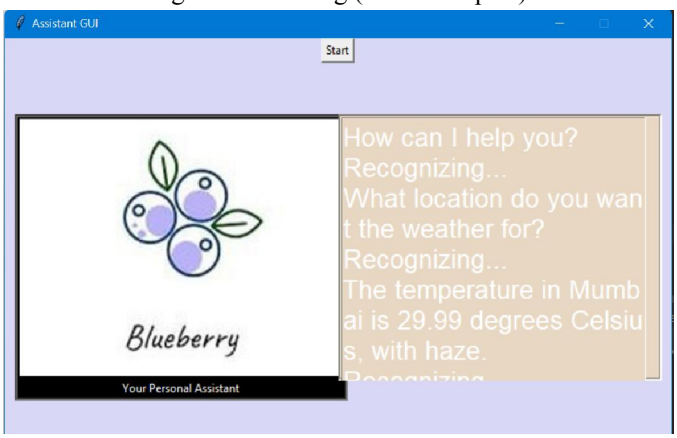


Fig 3: Functioning (weather report)



VII. IMPORTANT LIBRARIES /PACKAGE

1. Speech to Text: It is a program that converts audio to text. It is unable to understand anything you say.
2. Text Analyzing: Converted text is merely letters to a computer. Software is used to convert text so that computers can understand it. Because the computer understands the command, virtual assistants such as Siri interpret the text into a command for the computer. VPAs maps words to functions and parameters in order to produce a command that a computer can understand.
3. Speech Recognition Modules: For speech rearrangement, the system employs the Python-text-to-speech (pyttx3) package. It is used to transform user-supplied text. The pyttsx3 library in Python includes text-to-speech and speech-to-text converters. It encourages users to interact with the system.
4. API Calls: An API, or package interface, allows two programs to communicate with one another. Petite API refers to the connection that sends the request to the provider and then returns the provider's response.
5. Python Backend: The Python backend receives the users' text-based request and processes it to determine whether it involves an API call or data extraction. The system can then respond with the right response at any time.
6. Extraction of Data: It entails obtaining structured statistics from unstructured machine-readable information. In our proposed system, we have provided the relevant data that the user requested.
7. Text-To-Speech Module: This tool is extremely useful for persons who struggle with interpretation. Text-to-speech engines in a variety of languages, dialects, and complicated vascular architectures can be developed using third-party libraries.
8. Date and time: The Datetime package is used to display the date and time. Python already has a datetime module built in.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 1, June 2023

9. Wikipedia: We used the Wikipedia module in our project to get more information from Wikipedia or to run a Wikipedia search because, as we all know, Wikipedia, like GeeksforGeeks, is a tremendous and massive repository of knowledge. To install this Wikipedia module, run pip install Wikipedia.
10. Web Browser: This Python module is used to search the internet.
11. OS: The OS module in Python provides tools for interacting with the operating system. The fundamental utility modules for Python include OS. This module enables the use of operating system-specific functions.
12. Pyjokes: Pyjokes is an online joke collection tool. Pyjokes is included in our project since it contains jokes. It's extremely intriguing. Pyjokes is the one-line joke that makes our project more interesting.
13. Pyaudio: PortAudio is a C++ library that interfaces with audio drivers across several platforms. PyAudio is a suite of PortAudio Python bindings.

VIII. CONCLUSION

In conclusion, the construction of a voice-based virtual assistant using Python is an exciting area of research that has the potential to greatly improve user experiences in a wide range of situations, from smart homes to offices and beyond. A virtual assistant can enable people to engage with technology in a more intuitive and seamless manner by harnessing the capability of speech recognition and natural language processing, eliminating the need for physical interfaces such as keyboards or touch screens. Furthermore, by adding functions like weather detection, sentiment analysis, reminders, and appointment booking, the virtual assistant may provide a wide range of services and functionality that can help customers manage their daily chores more efficiently. We described a Python-based Voice Assistant in this paper. This assistant is currently application-based and performs basic activities like as weather updates, music streaming, Wikipedia searches, opening desktop apps, and so on. The current system's functionality is limited to solely working with applications. We discussed about Personal Virtual Assistant for Windows Using Python before. Virtual assistants have the freedom to contract for only the services they require. We create virtual assistants in Python for all Windows versions, just as Alexa, Cortana, Siri, and Google Assistant. Virtual Personal Assistants are an efficient method to manage and organize your time.

Virtual personal assistants are also more dependable than human assistants.

REFERENCES

[1] Vishal Kumar Dhanraj, Lokesh kriplani, Semal Mahajan, ”Research Paper on Desktop Voice Assistant” International Journal of Research in Engineeringand Science, Volume 10 Issue 2, February 2022.

[2] Prof. Suresh V. Reddy, Chandresh Chhari, Prajwal Wakde, NikhiKamble, ”Review on Personal Desktop Virtual Voice Assistant using Python” International Advanced Research Journal in Science, Engineering and Technology, Vol. 9 Issue 2, February 2022.

[3] Nivedita Singh, Dr. Diwakar Yagyasen, Mr. Surya Vikram Singh, Gaurav Kumar, Harshit Agrawal, ”Voice Assistant Using Python” International Journal of Innovative Research in Technology, Volume 8 Issue 2, July 2021, ISSN: 2349-6002.

[4] Edwin Shabu, Tanmay Bore, Rohit Bhatt, Rajat Singh,”A Literature Review on Smart Assistant” International Research Journal of Engineering and Technology (IRJET), Volume: 08 Issue: 04, April 2021.

[5] A. Sudhakar Reddy M, Vyshnavi, C. Raju Kumar, and Saumya, ”Virtual Assistant using Artificial Intelligence and Python” Journal of Emerging Technologies and Innovative Research (JETIR), Volume 7 Issue 3, March 2020, ISSN-2349-5162.

[6] Dr. Jaydeep Patil, Atharva Shewale, Ekta Bhushan, Alister Fernandes, Rucha Khartadkar, ”A Voice Based Assistant Using Google Dialogflow and Machine Learning” International Journal of Scientific Research in Science and Technology Volume 8 Issue 3, May 2021.

[7] Xin Lei, Andrew Senior, Alexander Gruenstein, Jeffrey Sorensen, “Accurate and compact large vocabulary speech recognition on mobile devices,” in INTERSPEECH. 2013, pp. 662–665, ISCA