VOICE ASSISTANT WITH

EMOTION

RECOGNITION

Submitted in partial fulfillment of the

requirements for the award of

Bachelor of Engineering degree in Computer Science and Engineering

By

SREENIVASAN MANIKATAN (Reg.No - 39110963)

SRAVANA SRIKANTH S (Reg.No - 39110985)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY (DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE

JEPPIAAR NAGAR, RAJIV GANDHISALAI, CHENNAI - 600119

APRIL - 2023

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

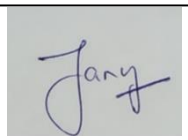
This is to certify that this Project Report is the bonafide work of **Sreenivasan Manikantan (Reg.No - 39110963) and SRAVANA SRIKANTH S (Reg.No - 39110985)** who carried out the Project Phase-2 entitled **"VOICE ASSISTANT WITH EMOTION RECOGNITION"** under my supervision from January 2023 to April 2023.

Internal Guide

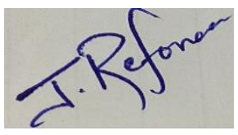
Dr. A. MARY POSONIA M.E, Ph.D.

Head of the Department
Dr. L. LAKSHMANAN, M.E,Ph.D.

Submitted for Viva voce Examination held on 24-04-2023



Internal Examiner



External Examiner

DECLARATION

I, "**SREENIVASAN MANIKANTAN (Reg.No- 39110963)",** hereby declare that the Project Phase-2 Report entitled **"VOICE ASSISTANT WITH EMOTION RECOGNITION "** done by me under the guidance of **Dr. A. MARY POSONIA, M.E, Ph.D** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering**.

DATE: 24-04-23

PLACE: Chennai **SIGNATURE OF THE CANDIDATE**

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph. D**, **Dean**, School of Computing, **Dr. L. Lakshmanan M.E., Ph.D.,** Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr.A. MARY POSONIA M.E.,Ph.D,**for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my phase-2 project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

An intelligent virtual assistant (IVA) or intelligent personal assistant (IPA) is a software agent that can perform tasks or services for an individual based on commands or questions. The term "chatbot" is sometimes used to refer to virtual assistants generally or specifically accessed by online chat. In some cases, online chat programs are exclusively for entertainment purposes. Some virtual assistants are able to interpret human speech and respond via synthesized voices. Users can ask their assistants questions, control home automation devices and media playback via voice, and manage other basic tasks such as email, to-do lists, and calendars with verbal commands. A similar concept, however with differences, lays under the dialogue systems.

As of 2017, the capabilities and usage of virtual assistants are expanding rapidly, with new products entering the market and a strong emphasis on both email and voice user interfaces. Apple and Google have large installed bases of users on smartphones. Microsoft has a large installed base of Windows-based personal computers, smartphones and smart speakers. Amazon has a large install base for smart speakers. Conversica has over 100 million engagements via its email and SMS interface intelligent virtual assistants for business.

The Most famous application of iPhone is “SIRI” which helps the end user to communicate end user mobile with voice and it also responds to the voice commands of the user. Same kind of application is also developed by the Google that is “Google Voice Search” which is used for in Android Phones.

TABLE OF CONTENTS

LIST OF ABBREVATION

S.NO	ABBREVATION
1	AI - Artificial Intelligence
2	API - Application Programming Interface
3	ASR - Automatic Speech Recognition
4	CNN - Convolutional Neural Network
5	DL - Deep Learning
6	FER - Facial Expression Recognition
7	FFT - Fast Fourier Transform
8	JSON - JavaScript Object Notation
9	NLP - Natural Language Processing
10	RNN - Recurrent Neural Network
11	SDK - Software Development Kit
12	STT - Speech-to-Text
13	TTS - Text-to-Speech
14	XML - Extensible Markup Language

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In today's era almost all tasks are digitalized. We have Smartphone in hands and it is nothing less than having world at your fingertips. These days we aren't even using fingers.

We just speak of the task and it is done. There exist systems where we can say Text Dad, "I'll be late today." And the text is sent.

That is the task of a Virtual Assistant. It also supports specialized task such as booking a flight, or finding cheapest book online from various ecommerce sites and then providing an interface to book an order are helping automate search, discovery and online order operations.

Virtual Assistants are software programs that help you ease your day-to-day tasks, such as showing weather report, creating reminders, making shopping lists etc.
They can take commands via text (online chat bots) or by voice. Voice based intelligent assistants need an invoking word or wake word to activate the listener, followed by the command.

This project was started on the premise that there is sufficient amount of openly available data and information on the web that can be utilized to build a virtual assistant that has access to making intelligent decisions.

Voice assistants have become increasingly popular in recent years, with the rise of smart home devices, virtual assistants, and other IoT technologies. The market for voice assistants is expected to continue to grow rapidly, with more than 8 billion digital voice assistants projected to be in use by 2023, according to a report by Juniper Research.

Voice assistants offer several benefits over traditional interfaces, such as keyboards and touchscreens. They provide a hands-free and eyes-free interface, allowing users to interact with devices while performing other tasks. They also enable more natural

and intuitive interactions, as users can speak in a more conversational style, rather than having to type or tap specific commands.

The AI and NLP technologies used by voice assistants allow them to understand and respond to a wide range of voice commands, including complex and ambiguous phrases. Voice assistants can also adapt to users' speech patterns and preferences, improving their accuracy and effectiveness over time.

Voice assistants can be integrated into a wide range of devices and services, from smartphones and smart speakers to home automation systems and healthcare applications. They can be customized to respond to specific commands and provide personalized recommendations and services.

As voice assistants become more sophisticated and ubiquitous, they are expected to transform the way we interact with technology and the world around us. They have the potential to make our lives more convenient, efficient, and enjoyable, and to enable new forms of communication and collaboration.

For a project, a voice assistant can be used to create a hands-free and convenient user interface to interact with a particular device or system. The voice assistant can be integrated into the project's hardware and software to enable voice control and enhance user experience.

The popularity of voice assistants has grown rapidly in recent years, as more people have become accustomed to using voice commands to interact with digital devices. Voice assistants offer a convenient, hands-free way to access information and control devices, and they are increasingly being used in a wide range of settings, including homes, offices, and cars.

Overall, voice assistants represent a significant shift in how people interact with technology, and they are expected to continue to play an increasingly important role in the way we communicate and interact with digital devices in the future.

CHAPTER 2

LITERATURE SURVEY

Radio Rex was the first voice activated toy released in 1922. It was a wooden toy in the shape of a dog that would come out of its house when its name is called.

In 1952, Bell Labs presented "Audrey", the Automatic Digit Recognition machine. It occupied a six- foot-high relay rack, consumed substantial power, had streams of cables and exhibited the myriad maintenance problems associated with complex vacuum-tube circuitry. It could recognize the fundamental units of speech, phonemes. It was limited to accurate recognition of digits spoken by designated talkers. It could therefore be used for voice dialing, but in most cases push-button dialing was cheaper and faster, rather than speaking the consecutive digits.

Another early tool which was enabled to perform digital speech recognition was the IBM Shoebox voice-activated calculator, presented to the general public during the 1962 Seattle World's Fair after its initial market launch in 1961. This early computer, developed almost 20 years before the introduction of the first IBM Personal Computer in 1981, was able to recognize 16 spoken words and the digits 0 to 9.

The first natural language processing computer program or the chatbot ELIZA was developed by MIT professor Joseph Weizenbaum in the 1960s. It was created to "demonstrate that the communication between man and machine was superficial". ELIZA used pattern matching and substitution methodology into scripted responses to simulate conversation, which gave an illusion of understanding on the part of the program.

Weizenbaum's own secretary reportedly asked Weizenbaum to leave the room so that she and ELIZA could have a real conversation.

Weizenbaum was surprised by this, later writing: "I had not realized ... that extremely short exposures to a relatively simple computer program could induce powerful delusional thinking in quite normal people.

This gave name to the ELIZA effect, the tendency to unconsciously assume computer behaviors are analogous to human behaviors; that is, anthropomorphisation, a

phenomenon presents in human interactions with virtual assistants.
The next milestone in the development of voice recognition technology in Pittsburgh, Pennsylvania with substantial support of the United States Department of Defense and its DARPA agency, funded five years of a Speech Understanding Research program, aiming to reach a minimum vocabulary of 1,000 words. Companies and academia including IBM, Carnegie Mellon University (CMU) and Stanford Research Institute took part in the program.

The result was "Harpy", it mastered about 1000 words, the vocabulary of a three-year-old and it could understand sentences. It could process speech that followed pre-programmed vocabulary, pronunciation, and grammar structures to determine which sequences of words made sense together, and thus reducing speech recognition errors.

In 1986 Tangora was an upgrade of the Shoebox, it was a voice recognizing typewriter. Named after the world's fastest typist at the time, it had a vocabulary of 20,000 words and used prediction to decide the most likely result based on what was said in the past. IBM's approach was based on a hidden Markov model, which adds statistics to digital signal processing techniques. The method makes it possible to predict the most likely phonemes to follow a given phoneme. Still each speaker had to individually train the typewriter to recognize his or her voice, and pause between each word.

Birth of smart virtual assistants: 1990s–past

In the 1990s, digital speech recognition technology became a feature of the personal computer with IBM, Philips and Lernout & Hauspie fighting for customers. Much later the market launch of the first smartphone IBM Simon in 1994 laid the foundation for smart virtual assistants as we know them today.

In 1997, Dragon's Naturally Speaking software could recognize and transcribe natural human speech without pauses between each word into a document at a rate of 100 words per minute. A version of Naturally Speaking is still available for download and it

is still used today, for instance, by many doctors in the US and the UK to document their medical records.

In 2001 Colloquis publicly launched Smarter Child, on the platforms like AIM and MSN Messenger. While entirely text-based Smarter Child was able to play games, check the weather, look up facts, and converse with users to an extent.

The first modern digital virtual assistant installed on a smartphone was Siri, which was introduced as a feature of the iPhone 4S on 4 October 2011. Apple Inc. developed Siri following the 2010 acquisition of Siri Inc., a spin-off of SRI International, which is a research institute financed by DARPA and the United States Department of Defense. Its aim was to aid in tasks such as sending a text message, making phone calls, checking the weather or setting up an alarm. Over time, it has developed to provide restaurant recommendations, search the internet, and provide driving directions.

In November 2014, Amazon announced Alexa alongside the Echo.

In April 2017 Amazon released a service for building conversational interfaces for any type of virtual assistant or interface. Below figure represents various voice assistants that are embedded with our technologies like smart home devices etc.

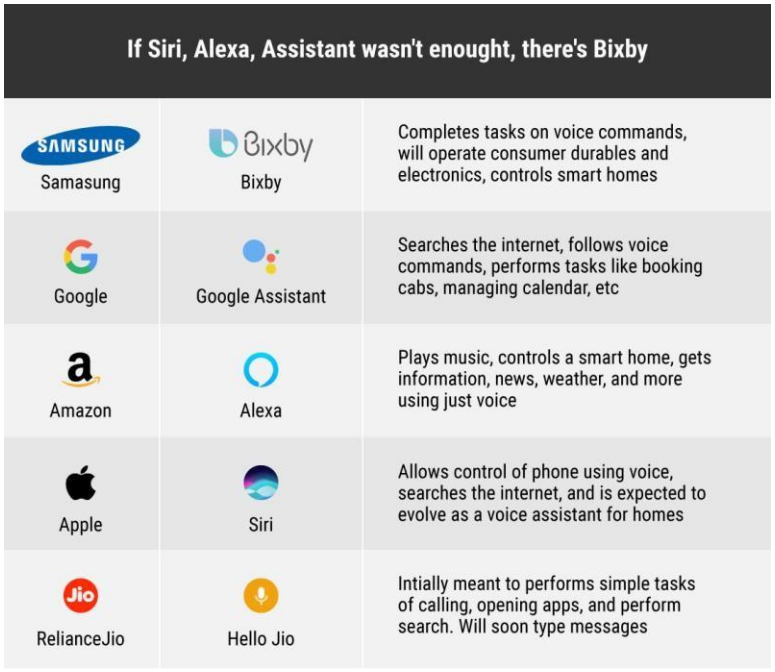


Fig - 2.1 voice Assistants

2.1 INFERENCES FROM LITREATURE SURVEY

There already exist several desktop virtual assistants. A few examples of current virtual assistants available in market are discussed in this section along with the tasks they can provide and their drawbacks.

SIRI from Apple

SIRI is personal assistant software that interfaces with the user through voice interface,
recognizes commands and acts on them. It learns to adapt to user's speech and thus improves voice recognition over time. It also tries to converse with the user when it does not identify the user request.
It integrates with calendar, contacts and music library applications on the device and also integrates with GPS and camera on the device. It uses location, temporal, social and task.
based contexts, to personalize the agent behavior specifically to the user at a given point of time.

Supported Tasks

- Call someone from my contacts list
- Launch an application on my iPhone
- Send a text message to someone
- Set up a meeting on my calendar for 9am tomorrow
- Set an alarm for 5am tomorrow morning
- Play a specific song in my iTunes library

Drawback

SIRI does not maintain a knowledge database of its own and its understanding

comes from the information captured in domain models and data models.

ReQall

ReQall is personal assistant software that runs on smartphones running Apple iOS or Google Android operating system. It helps user to recall notes as well as tasks within a location and time context.
It records user inputs and converts them into commands, and monitors current stack of user tasks to proactively suggest actions while considering any changes in the environment.
It also presents information based on the context of the user, as well as filter information to the user based on its learned understanding of the priority of that information.

Supported Tasks

- Reminders
- Email
- Calendar, Google Calendar
- Outlook
- Evernote
- Facebook, LinkedIn
- News Feeds

Drawback

Will take some time to put all of the to-do items in – you could spend more time putting the entries in than actually doing the revision.

2.2 OPEN PROBLEMS IN EXISTING SYSTEM

1. Lack of lingual knowledge

What makes speech recognition difficult is the lack of language training.
Companies often seem to overlook the fact that English is not the universal language. So expecting users from different geographies to have the same level of proficiency is unrealistic. In fact, 38% of users are hesitant to adopt voice technology because of AI's language coverage.
If you are trying to deploy your voice assistants in a location, the ASR will likely tank if not trained on specific language models of the region. And even when it is trained for the language, another challenge for ASR is the ability to differentiate between varying dialects and accents for more accurate interpretation.
For example, a user who needs groceries may say "Buy vege-table" to the voice assistant, pronouncing the word a bit differently than the widely accepted "veg-tible" – also the only one AI is familiar with. A poorly trained bot may mistake this input as "buy a veggie table" assuming the speaker wants to buy a table – Highly inaccurate!

2. Peripheral background sounds

Another top speech recognition problem that needs a solution is – noise. It is everywhere! And so, it becomes the job of the ASR solution to accurately catch the speech input through unwanted sounds. An ASR should be able to pick up the input's sound waves even from a distance in a room riddled with white noise and cross-talk. Echo, for example, also adds to the imprecision. Reflected sound waves from surfaces in the space distort the receptor's ability to process the actual input unerringly.

3. Low data reliability of ASR

What are the other challenges of speech recognition? Data privacy. While we are making progress in the field of AI, many users are still hesitant to use ASR bots to handle tasks that involve sensitive data and money. Data privacy is sovereign to users who wish to exercise some level of governance and transparency with their information.

PWC says that one of the three main reasons why users are scared to experiment with voice tech is simply a lack of trust. Where more than half of the users use their voice assistant to buy online, all of these purchases are trivial with low spending. And so, data concerns remain the challenges and issues businesses face in adopting speech recognition technology. Users don’t trust voice assistants as much, so businesses must be prepared to face reluctance in adoption from their market.

4.Costs and deployment

Implementing an ASR system needs a far-sighted vision. It’s a long game and not a change that occurs overnight. Bearing this in mind, you need to be prepared to handle the time, resources, and capital involved in building, testing, and deploying the system in the market. For example, the lack of visual elements makes designing interactive voice user interfaces (VUIs) more complex than designing UI for chatbots.

Another disadvantage of speech recognition can be that training language models take considerable time and expertise. Gathering enough language resources or effectively making do with the available ones may not come cheap. All in all, manual development would rain heavily on your pockets.

CHAPTER 3

3.1 FEASIBILITY STUDIES/RISK ANALYSIS OF THE PROJECT

1. ANDROID APP AVAILABILITY?

Yes, it is feasible to develop an android application in pure python.
In order to achieve this, I suggest you use BeeWare, which is just a **suite of python tools,** that work together very well and they enable you to develop platform native applications in python.

2. HOW IT WORKS

It works based on voice commands. With the usage of single voice commands things done easier and efficient way through coded AI assistant. Performing tasks with voice commands and improving it further for n number of tasks that are needed to attain and move forward through technologies. Innovations can be dealt in this way. Before knowing about this the assistant must be woken up with the wake-up function, thereby proclaims an intermediate cause or disturbance does not provide chunk or clumsy to assistant. Tasks are needed to define with its given time range. Excluding time range does not produce anything and provide negligible data. Although time range can be set manageable, it can't get the point or a conclusion about hour speech or vast range of query. The flow chart is attached in the given paper for better understanding of the workflow and better to understand.

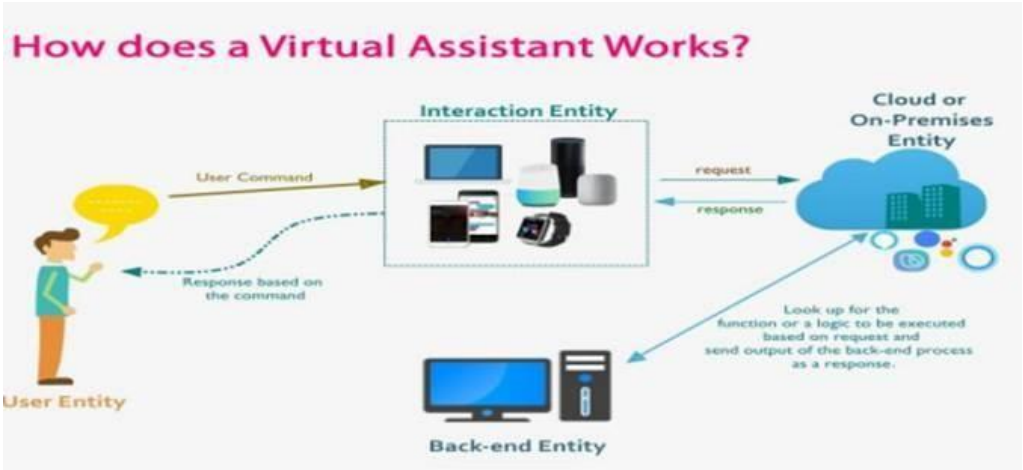


Fig – 3.1 base working model outline

Natural Language Processing (NLP) and speech recognition module helps to obtain the best outcome through labelled input voice data. By obtaining the audio from various user the nlp is trained and brought with its speech recognition module to perform the audio processing in any harsh condition and there by proceeding with necessary needs.

Emotion Recognition is to identify possessive emotion from a person and evaluate the need accordingly. It restrains the mood of approach and provide essential output required to balance the emotion by providing multiple tasks etc.

Briefcase is a tool for converting a Python project into a standalone native application. You can package projects for:

- Mac
- Windows
- Linux
- Android

In addition to the BeeWare tools, we'll need to have a JDK and Android SDK installed to test run your application.A good environment can be anything you are comfortable with be it a text editor and a command line, or an IDE, if you're looking for a good python IDE

BEST PYTHON TOOLS FOR DEVELOPING ANDROID APP

- Kivy
- PyQt5
- Chaquopy with Beeware.

3.2 SOFTWARE REQUIREMENTS SPECIFICATION

DOCUMENT HARDWARE:

• Pentium-pro processor or later.
• RAM 512MB or more.

SOFTWARE:

- Windows 7(32-bit) or above.
- Python 3.5 or later
- Chrome Driver
- Selenium Web Automation

3.3 SYSTEM USE CASE

Voice Assistant are designed to make our lives easier and more convenient by providing hands-free control over our devices and services. Here are some detailed use cases of voice assistants:

1. **Smart Home Control**: One of the most common uses of voice assistants is for controlling smart home devices such as smart lights, thermostats, and security systems. With voice commands, users can easily turn on or off the lights, adjust the temperature, or lock the doors without having to manually interact with the devices.

2. **Information Retrieval**: Voice assistants can quickly retrieve information on any topic that the user requires. For instance, users can ask for weather updates, news headlines, or directions to a nearby restaurant. Voice assistants can also answer general knowledge questions and provide definitions for unfamiliar terms.

3. **Hands-Free Communication**: Voice assistants can be used for hands-free communication, such as sending text messages, making phone calls, or sending emails. This can be particularly useful when driving or when the user's hands are otherwise occupied.

4. **Personal Reminders**: Voice assistants can be used to set reminders for tasks and appointments. Users can simply say, "Hey, Google, remind me to call the dentist at 2 pm," and the voice assistant will set a reminder for that time. This

is a great way to stay organized and on top of important tasks.

5. **Entertainment**: Voice assistants can also be used for entertainment purposes. Users can ask for music or podcasts to be played, or even play games with the voice assistant. This is a great way to pass the time and unwind after a long day.

6. **Accessibility**: Voice assistants can be especially helpful for people with disabilities or mobility issues. They can provide a hands-free way to control devices and access information without the need for physical interaction.

7. **Smart Home Control**: One of the most common uses of voice assistants is for controlling smart home devices such as smart lights, thermostats, and security systems. With voice commands, users can easily turn on or off the lights, adjust the temperature, or lock the doors without having to manually interact with the devices.

8. **Information Retrieval**: Voice assistants can quickly retrieve information on any topic that the user requires. For instance, users can ask for weather updates, news headlines, or directions to a nearby restaurant. Voice assistants can also answer general knowledge questions and provide definitions for unfamiliar terms.

9. **Hands-Free Communication**: Voice assistants can be used for hands-free communication, such as sending text messages, making phone calls, or sending emails. This can be particularly useful when driving or when the user's hands are otherwise occupied.

10. **Personal Reminders**: Voice assistants can be used to set reminders for tasks and appointments. Users can simply say, "Hey, Google, remind me to call the dentist at 2 pm," and the voice assistant will set a reminder for that time. This is a great way to stay organized and on top of important tasks.

11. **Entertainment**: Voice assistants can also be used for entertainment purposes. Users can ask for music or podcasts to be played, or even play games with the voice assistant. This is a great way to pass the time and unwind after a long day.

12. **Accessibility**: Voice assistants can be especially helpful for people with disabilities or mobility issues. They can provide a hands-free way to control devices and access information without the need for physical interaction.

Overall, the use cases of voice assistants are diverse and wide-ranging, and the possibilities for their applications are continually expanding. As voice assistants become more advanced and integrated with various devices and services.

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

Looking at the disadvantages of all the above methodologies used in previous systems, the most common point that pops up is most of these Artificial intelligence systems implemented this problem consist of some set of works that can be assigned in built or by default only some of them will be executed as a matter of fact.

Deep down with the help of Human Intelligence these AI can be modified in various ways like to fetch, create, analyze, modify, delete, notify, Recognize etc. Currently only some of the devices can achieve almost everything from the given command. Most of them can be performed better even in worst case scenario by updating AI. Here we make use of this in our way which will be similar to other but the updated model with a rapid grasp in technology.

4.1 SELECTED METHODOLOGY OR PROCESS MODEL

Virtual assistants work via:

SMS text, e-mail or other text-based communication channel, for example Conversica's intelligent virtual assistants for business.

By taking and/or uploading images, as in the case of Samsung Bixby on the Samsung Galaxy S8 as shown in figure 4.1.

Some virtual assistants are accessible via multiple methods, such as Google Assistant via chat on the Google Allo and Google Messages app and via voice on Google Home smart speakers.

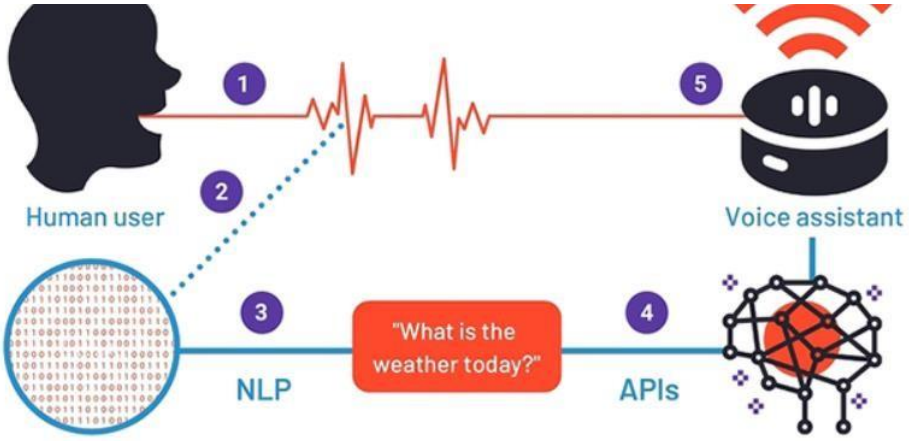


Fig – 4.1 conversations

Virtual assistants use natural language processing (NLP) to match user text or voice input to executable commands. Many continually learn using artificial intelligence techniques including machine learning. Some of these assistants like Google Assistant(which contains Google Lens) and Samsung Bixby also have the added ability to do image processing to recognize objects in the image to help the users get better results from the clicked images.

4.2 ARCHITECTURE / OVERALL DESIGN OF PROPOSED SYSTEM

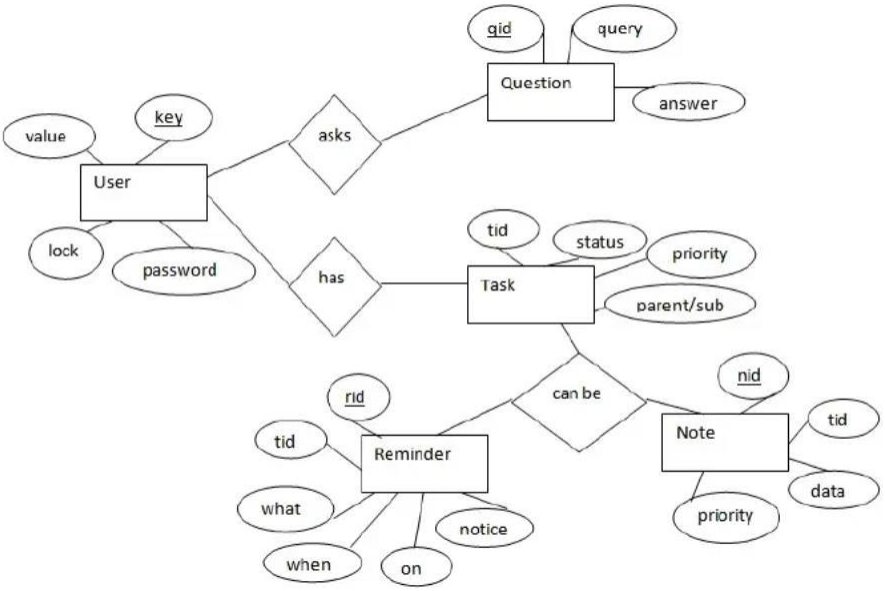


Fig – 4.2 complex system architecture

Single user can ask multiple questions. Each question will be given ID to get recognized along with the query and its corresponding answer. User can also be having n number of tasks. These should have their own unique id and status i.e. their current state. A task should also have a priority value and its category whether it is a parent task or child task of an older task.

Finally once the response is identified, output is generated from simple text to speech conversion using text to speech (TTS). The above tasks can be performed using certain methodologies in which each technique has its own functionality and different operations to be performed. Each technique has different process logic to be

executed.

4.3 DESCRIPTION OF SOFTWARE FOR IMPLEMENTATION AND TESTING PLAN OF THE PROPOSED MODEL/SYSTEM

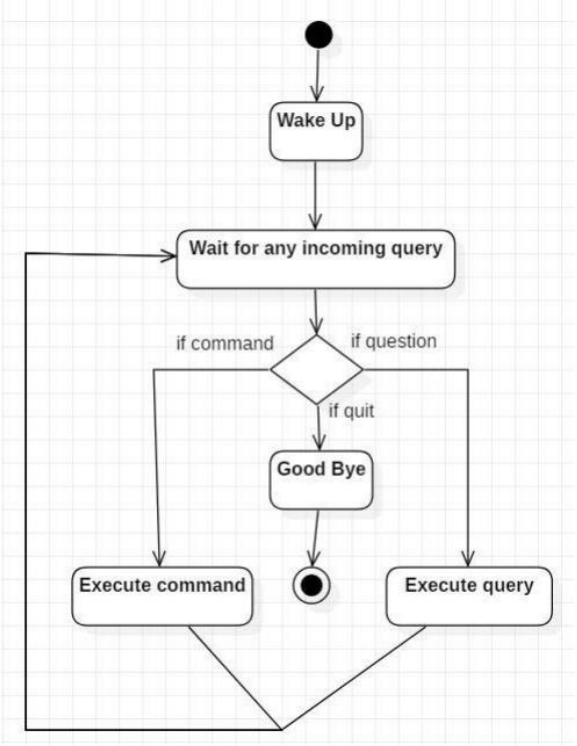


Fig – 4.3 Implementation

Initially, the system is in idle mode. As it receives any wake-up call it begins execution. The received command is identified whether it is a questionnaire or a task to be performed. Specific action is taken accordingly. After the Question is being answered or the task is being performed, the system waits for another command. This loop continues unless it receives quit command. At that moment, it goes back to sleep.

4.4 PROJECT MANAGEMENT PLAN

This system communicates with the user and through using similar methods natural language recognition tries to provoke social conversation with user.

The software aims to give personal companionship and to replicate human interaction

as accurately as possible with the assistance of algorithms designed to help the program learn from its inputs.

The overall system design consists of following phases:

1. Data collection in the form of speech.
2. Voice analysis and conversion to text
3. Data storage and processing
4. Generating speech from the processed text.

System Architecture In first phase, the data is collected in the form of speech and stored as an input for the next phase for processing. In second phase, the input voice is continuously processed and converted to text using speech to text (STT).

In next phase the converted text is analyzed and processed using Python Script and NLP techniques to identify the response to be taken against the command. Finally once the response is identified, output is generated from simple text to speech conversion using text to speech (TTS). Problems with Existing System: Despite the various benefits provided by speech recognition, the system is also plagued with limitations. By implication the development of speech recognition applications also inherits these limitations.

The existing Voice Assistants use set of pattern recognition techniques of misinterpretations, Time, costs and productivity, User accents. They operate based on the functions provided that requires internet.

Emotions are classified in several types. So for the system to analyze human emotions here we use different facial images containing various emotions that are predictable and trained under set of proper algorithms for better visualizing and thereby applying the technique to the user provided facial data and predicting it.

Emotion recognition is a technology by which the system tries to know the mindset of the person who is using it. Emotion recognition is one of the key areas which this project tries to focus on by using this module the focus is to find the current mood of

our client and to suggest things based on it to make him feel more satisfied with the working of our assistant the module used behind this is advanced image processing techniques which makes use of Convolutional Neural Network which is used to breakdown the image into various small pixels and analyze these pixels for a better understanding of the mood.

HUMAN EMOTION PREDICTION

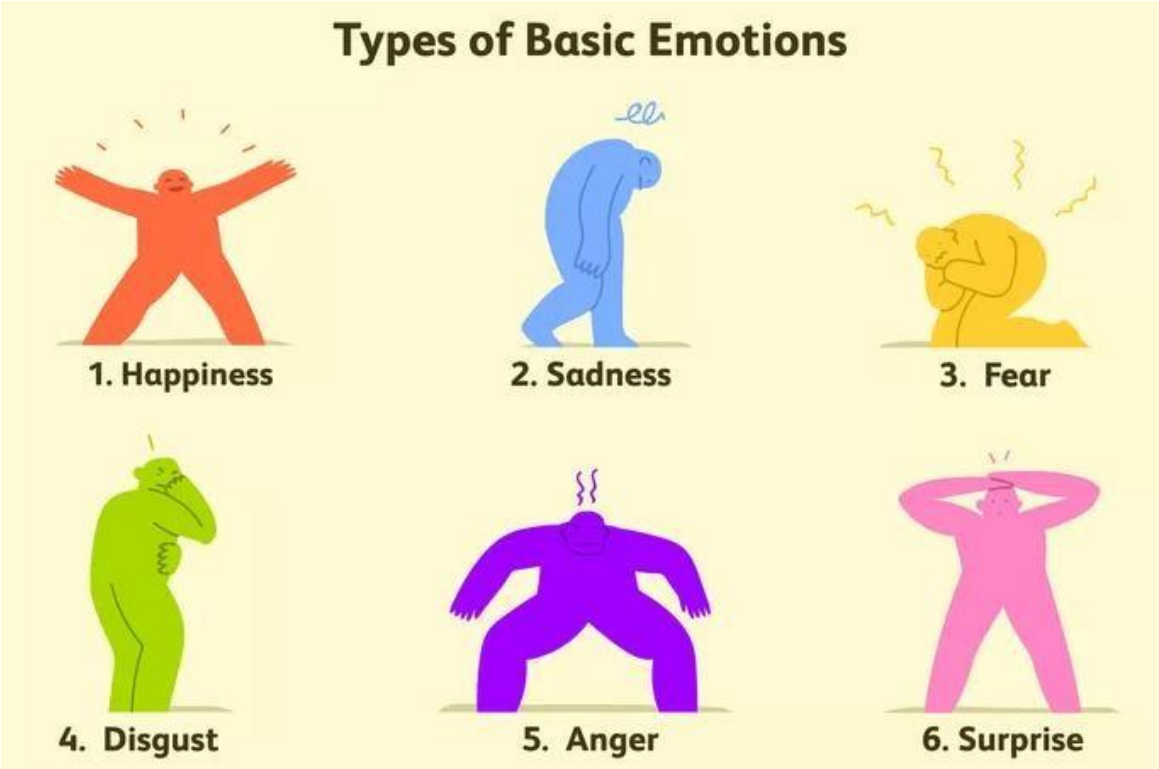


Fig – 4.4 Basic Emotions

Humans possess different kinds of emotions out of those some of them are represented in the above figure.

Voice assistant is capable of recognising human emotions in different scenarios.

Either through audio or video as input it can predict the emotions.

Through audio signals the pitch, tone, wave length of a person will be identified and the audio will be refrained by method cleaning which enhances the audio with the removal of background noise and provide the crystal clear output.

With the help of facial expression it can easily identify the emotions and predict what

category it belong to.
According to these emotions it can engage with music or provide essential support like refreshment, condolence etc.
This is the essential feature provided to the assistant . Approximately 80% of the emotions provided by humans can be detected.

4.5 TRANSITION/ SOFTWARE TO OPERATIONS PLAN

So, let's have a brief of the new updated version of the voice assistant. Instead of pattern recognition technique which has been used in previous models, we use Natural Language Processing (NLP) techniques to recognize the text which is context based rather the usual pattern based. This Operates in online as well as offline mode. System application runs on offline mode, whereas web based operations run on online mode. Data is Stored in Application itself, rather than cloud which reduces Time and Space Complexity. It even reduces the economic cost due to reducing high bundles of data usage.

Logic Personal Voice Assistant is developed as a desktop application with the help of Natural Language Processing which helps to send messages and use various built-in systems based and web-based applications using voice commands. The Voice Assistant performs basic operations such as controlling computer tasks and operations, asking for temperature, humidity, date, time, and year. Adding, reading and deleting notes using voice commands and playing YouTube videos on demand.

The above tasks can be performed using certain methodologies in which each technique has its own functionality and different operations to be performed. Each technique has different process logic to be executed.
Techniques that are to be implemented/required:
1. Speech Recognition
2. NLP
3.Threadin
g

Virtual assistants use natural language processing (NLP) to match user text or voice input to executable commands. When a user asks a question to personal assistant to perform a task, then natural language audio signal is converted into executable command or digital data that can be analyzed by the software. Then this data is compared with a data of the software to find a suitable answer.

Virtual Assistant is used to run machines on your own commands. For making virtual assistant we use some python installer packages like Speech recognition, gTTS, pipwin, etc. Speech recognition is the process of converting audio into text. This is commonly used in voice assistants
Python provides an API called Speech Recognition to allow us to convert voice or audio command into text for further processing. By above diagram, firstly users give the command to the interaction entities like laptop, PC's this interaction entities listen the command and recognize it.

For further analyzing process compare this command with cloud in which we already store data. After matching request the output is generated in the text as well as voice form if the request is match with cloud data.

For emotion recognition required libraries and set of compataible devices must be installed to perform required tasks. There by we already have the enough source and outcomes for the relevant data. We gather the different functions using thoughts, advice, ideas and put it into action.

4.6 FINANCIAL REPORT ON ESTIMATED COSTING

Certain technologies like microphone speakers are required for processing since the resources are available quietly and the code can be work in hassle free environment. There are nearly 95 hours of the estimated cost of duration to complete the whole project and it is the rough estimation provided within the given limit of time. For sustainable dataset we preferred to add the require knowledge and effort by applying the fer 2013 facial images for the project that made look easier in a consistent time.

CHAPTER 5

IMPLEMENTATION DETAILS

5.1 Development and Deployment Setup

Developing and deploying a voice assistant (VA) requires a combination of hardware and software tools, along with expertise in natural language processing (NLP) and artificial intelligence (AI). Here's an overview of the development and deployment setup for VA:

1. **Hardware Requirements:** To develop and deploy a voice assistant, you need a hardware setup that can support the software and services required. This includes a microphone or speaker, a powerful processor, and memory for data storage.

2. **Software Requirements:** Developing a voice assistant requires a combination of software tools and services, including a programming language, machine learning frameworks, and APIs for speech recognition and text-to-speech conversion. Popular programming languages for voice assistant development include Python, Machine learning frameworks like TensorFlow, and Keras are commonly used for building AI models.

3. **NLP and AI Expertise**: Developing a voice assistant requires expertise in NLP and AI, which are the key technologies behind voice assistants. NLP is used to understand and interpret natural language commands and queries, while AI is used to create personalized experiences for users.

4. **Testing and Optimization**: Before a voice assistant can be released to the public, it needs to be tested and optimized for performance. This includes testing its speech recognition and text-to-speech conversion capabilities, as well as ensuring that it can handle a wide range of user queries and commands.

5. **Continuous Improvement**: Once a voice assistant is deployed, it requires ongoing maintenance and continuous improvement. This includes collecting feedback from users, analyzing usage data, and updating the AI models to improve their accuracy and relevance.

6. **Data Collection and Preprocessing**: One of the most critical steps in developing a voice assistant is collecting and preprocessing the data. This involves gathering a large dataset of voice commands, queries, and natural language sentences that the voice assistant will be trained on. The data must be cleaned, normalized, and structured in a way that the AI models can understand.

7. **Model Training**: After collecting and preprocessing the data, the next step is to train the AI models. This involves using machine learning algorithms to analyze the data and build models that can recognize and interpret natural language queries and commands. The AI models are typically trained using supervised learning techniques, where they are fed a large dataset of labeled examples and learn to recognize patterns in the data.

8. **Model Deployment**: Once the AI models have been trained, they need to be deployed on a platform that can support their functionality. This involves integrating the models with the speech recognition and text-to-speech conversion APIs, as well as the backend servers that process the user's queries and commands.

Overall, developing and deploying a voice assistant is a complex process that involves a combination of hardware and software tools, AI expertise, and a thorough understanding of natural language processing. The process requires careful planning, testing, and optimization, but the potential benefits of having a voice assistant for businesses and consumers are significant.

Below are set of major function's that are implemented and given below in a tabular format. Table- 1.1. This table represents how the output is evaluated for specific query and result mentions it.

Query	Function/Module	Result
news	Apikey+ news	News based on different sectors.
Alarm	datetime	Rings alarm
Today's date, time	datetime	Respective date time at that region
Search engine	Google, Wikipedia, you tube	Produce output with through provided search engine
weather	Apikey+ location	Provide weather report for given query about a place.
"Based on provided input from a query"	wolframalpha+ Apikey+ query	Produces output for mathematical calculation, complex problems in science, GK etc.
Open/ close	Pyautogui, os	Open/close application in OS.
Change wallpaper	Ctypes, os	Replace current wallpaper from a specified folder of

		images.
Recycle bin	winshell	Empty items in the recycle bin.
songs	Startfile,os, taskkill	Can play or stop the song.
volume	Pynput.keyboard	Increase/decrease volume

Table – 1.1

Query & Output

From the given data after calculating with 100 epochs, we got the accuracy of trained set is 94.1, validation set is 61.4 in accuracy graph. and the loss of trained set is 0.12, validation set is 1.64 in loss graph. To decrease loss or validation, the solution is to increase epochs, number of layers, hidden units, CNN, validation steps in generator. One of the major drawbacks is this model requires best graphical processor unit processor to validate and need more duration for estimating.

5.2 ALGORITHMS

Here are some of the common algorithms used in voice assistant:

Speech Recognition: Speech recognition algorithms are used to convert spoken words into text. The algorithms analyze the sound waves and use statistical models to recognize patterns in the speech. Popular speech recognition algorithms include Hidden Markov Models (HMMs), Gaussian Mixture Models (GMMs), and Deep Neural Networks (DNNs).

Natural Language Understanding (NLU): NLU algorithms are used to analyze the meaning behind natural language queries and commands. The algorithms use machine learning techniques to identify the intent of the user's query and extract key information. NLU algorithms use techniques like semantic analysis, syntactic analysis, and sentiment analysis to understand the user's intent and context.

Machine Learning: Machine learning algorithms are used to train the AI models that

power the voice assistant. These algorithms use supervised learning techniques to learn from labeled datasets and unsupervised learning techniques to identify patterns in unstructured data. Popular machine learning algorithms used in voice assistant development include decision trees, random forests, support vector machines, and deep neural networks.

Convolutional Neural Networks (CNNs): CNNs are deep learning algorithms that are often used for FER. The algorithm analyzes the pixels in an image and uses multiple layers of processing to identify facial features and detect emotions.
A typical CNN consists of several layers, including convolutional layers, pooling layers, and fully connected layers. Here's a brief overview of each layer:

1. ***Convolutional Layers:*** Convolutional layers are the primary building blocks of a CNN. Each layer consists of a set of filters that are applied to the input image to extract features. The filters slide across the image and perform a mathematical operation known as convolution. The output of the convolutional layer is a set of feature maps that represent the learned features.

2. ***Pooling Layers:*** Pooling layers are used to reduce the spatial size of the feature maps by down sampling the data. The most common pooling technique is max pooling, which selects the maximum value within a given region of the feature map. By reducing the size of the feature maps, pooling helps to reduce the computational complexity of the network and prevent overfitting.

3. ***Fully Connected Layers:*** Fully connected layers are used to perform the final classification of the image. These layers take the flattened feature maps from the previous layers and pass them through a set of neurons, which perform a series of mathematical operations to classify the image.

From the dataset after calculating with 100 epochs, we got the accuracy of trained set is 94.1, validation set is 61.4 in accuracy graph. and the loss of trained set is 0.12,

validation set is 1.64 in loss graph. To decrease loss or validation, the solution is to

increase epochs, number of layers, hidden units, CNN, validation steps in generator. One of the major drawbacks is this model requires best graphical processor unit processor to validate and need more duration for estimating.

Epochs	Accuracy	Validation
20	0.6455	0.5919
40	0.8114	0.6244
60	0.8973	0.6246
80	0.9357	0.6242
100	0.9490	0.6263
	Result **94%**	

Table – 2.1

Accuracy vs Validation Accuracy

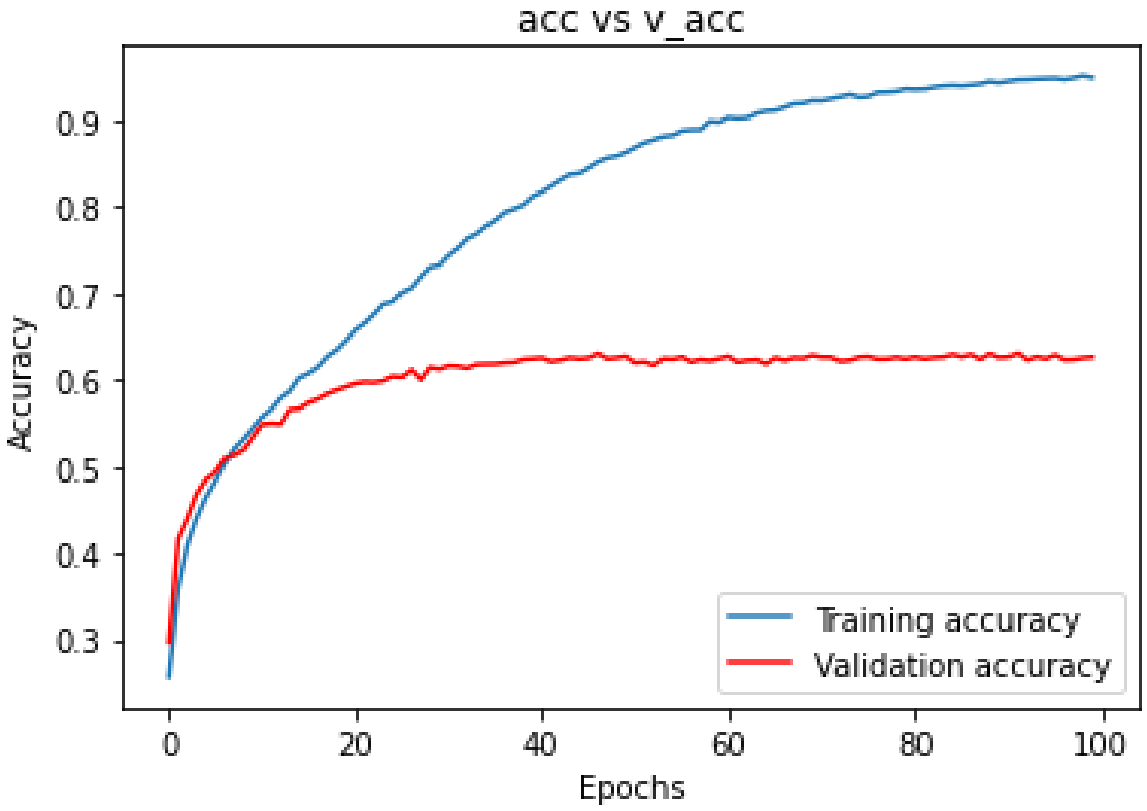


Fig – 5.1 Accuracy Graph

The figure 2.1, 2.2 shows about accuracy of dataset being trained and validating it (2.1), also the loss is predicted in 2.2. note that the epochs taken here are 100 and time took to neural network is nearly 4 hours (rough estimation, might varies). Table 2.1 represents convolutional neural network trained model at different epochs by validating it with images the accuracy and loss is generated. At 100th epoch the accuracy is 94% and loss is 14%. Fig 2.2 shows Loss and validation loss.

Epochs	Loss	Validation
20	0.9570	1.0924
40	0.5222	1.1308
60	0.2870	1.3062
80	0.1852	1.4467
100	0.1462	1.5940
	Result **14%**	

Table – 2.2

Loss vs Validation Loss

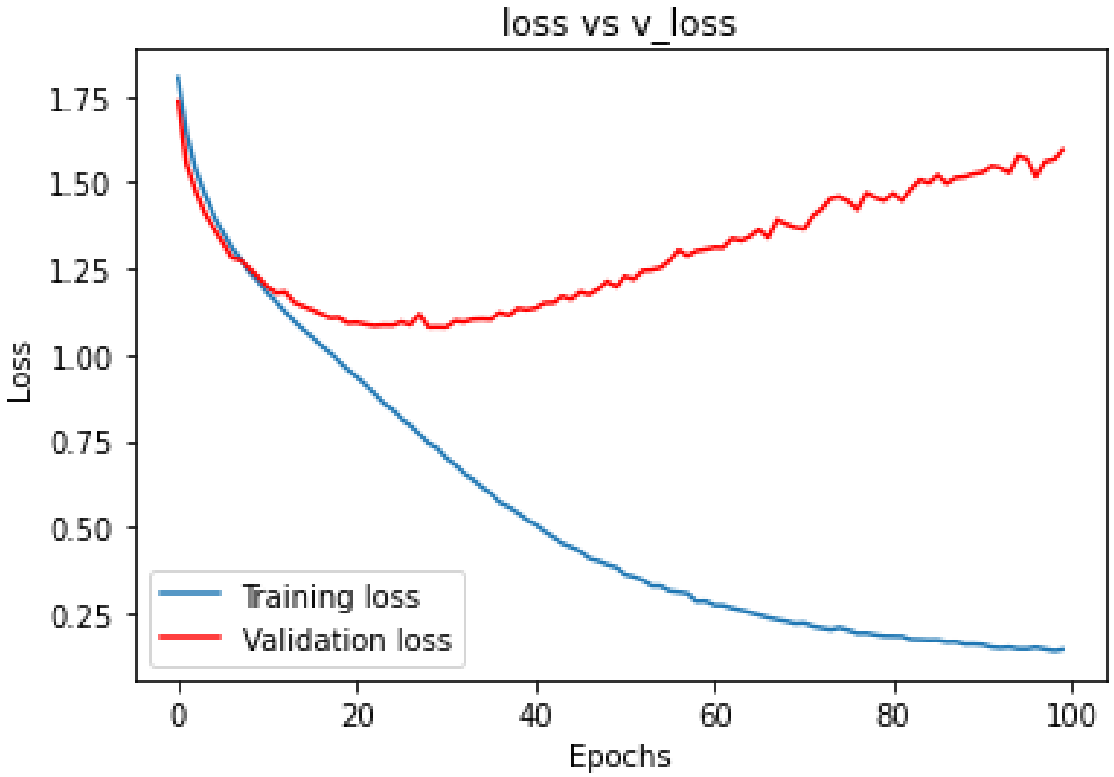


Fig – 5.2 loss graph

5.3 TESTING

Testing of a voice assistant involves evaluating its functionality, usability, and performance. Here are some key aspects to consider when testing a voice assistant:

1. **Functionality Testing**: Functionality testing involves ensuring that the voice assistant performs the functions it is designed to do. This includes testing features such as voice recognition, natural language processing, and response generation. Testers should check if the voice assistant can understand user queries and respond with the appropriate information or action.

2. **Usability Testing**: Usability testing involves evaluating how easy and intuitive the voice assistant is to use. Testers should ensure that the voice assistant interface is user-friendly, and that users can navigate through the different features without any difficulties. They should also evaluate how well the voice assistant responds to different accents, pronunciations, and speaking styles.

3. **Performance Testing**: Performance testing involves evaluating the voice assistant's response time, reliability, and scalability. Testers should ensure that the voice assistant responds quickly and accurately to user queries, and that it can handle large volumes of traffic without slowing down or crashing. It involves with implementation of emotion recognition with proper testing with the help of dataset and enhance the projected output.

Testing of a voice assistant should be done in multiple environments, such as quiet rooms, noisy environments, and different accents and languages. This will help ensure that the assistant performs well in real-world situations.

CHAPTER 6

RESULTS AND DISCUSSION

The results and discussion for a voice assistant project will depend on the specific goals and objectives of the project. However, some common areas of evaluation and discussion for a voice assistant project are:

1. **Accuracy and effectiveness**: One key measure of success for a voice assistant is its accuracy and effectiveness in understanding and responding to user voice commands. This can be evaluated through user testing and feedback, as well as automated metrics such as word error rate (WER) and intent match rate (IMR).

2. **User experience**: Another important factor in evaluating a voice assistant is the user experience. This includes factors such as ease of use, responsiveness, and naturalness of the voice prompts and responses. User surveys and feedback can be used to assess the user experience.

3. **Integration and compatibility**: A voice assistant project may also need to be evaluated on its integration with other hardware and software systems, as well as its compatibility with different languages and accents. Testing and evaluation of these aspects may involve integration testing, compatibility testing, and localization testing.

4. **Security and privacy**: As voice assistants often involve personal and sensitive information, it is important to evaluate the security and privacy aspects of the project. This may involve testing for vulnerabilities, ensuring secure data storage and transmission, and complying with data protection regulations.

5. **Performance and scalability**: Finally, a voice assistant project may need to be evaluated on its performance and scalability. This includes factors such as response time, resource usage, and the ability to handle a large number of concurrent users. Performance testing and evaluation can help identify

bottlenecks and optimize the system for better scalability.

Another important topic to consider in the results and discussion for a voice assistant project is the impact of the voice assistant on the business or organization. This includes factors such as cost savings, efficiency gains, and increased customer satisfaction.

For example, a voice assistant in a call center may lead to reduced call times and increased customer satisfaction, as customers are able to quickly and easily resolve their issues using voice commands. Similarly, a voice assistant in a retail store may lead to increased sales and customer engagement, as customers are able to ask questions and receive personalized recommendations using voice commands.

The results and discussion should also consider any challenges or limitations that were encountered during the development and deployment of the voice assistant. These may include technical challenges, such as integrating with existing systems, as well as organizational challenges, such as resistance from employees or customers.
The discussion of the results for a voice assistant project should address the strengths and weaknesses of the system, as well as areas for improvement and future development. It should also consider the impact of the voice assistant on the user experience and the overall goals of the project. Recommendations for further development, optimization, and integration can be discussed, based on the evaluation results.

CHAPTER 7

7.1 CONCLUSION

At the end of this project, I would like to conclude that AI is the future, and the voice assistants are the most basic part of AI in future. It would be engaged in sectors being taken over by these AI bots so that the load on the humans is reduced and the productivity of the activities in which people are involved in increases which might be unbeneficial for humankind.

Also because of these assistants there is no need to type something or even try opening something, just have to open our device and call out the voice assistant's name then it will do whatever the work that have on the internet. just have to provide the voice command of the required activity that has to be done and the assistant will take care of everything right from understanding our input to giving us the required output.

In this voice assistant there is this feature called emotion recognition which helps in recognizing the mood of the user and also giving out the proper suggestions based on that so for instance if someone is sad then it automatically recognizes that and suggests them things to enlighten their mood thereby can do a wholesome lot of things by using this like reading out news headlines giving weather updates and many more things which people do by typing out on internet without this assistant.

AI is a tool which is made to make the life of a human easy as the computer is doing things that a human can do as intelligently as a human so that the life of humans becomes very easy as machine can take care of most of the things as perfectly as humans so there is less amount of work that the humans have to consider so the productivity increases and the portion of the work people have to concentrate on is also reduced. But always have to be very careful about one thing in future that, make sure not to make them (refers to AI) so advance that it even might start control over mankind which eradicate everything and involves in every action.

7.2 FUTURE WORK

For the future work here are some of the useful ideas that can roll out voice assistant by further needs.

- Application accessing interface.
- Voice embedded service center.
- Futuristic car systems.
- Embedded tech.
- Education
- Healthcare
- Smart home devices
- Voice enabled tracking system etc.

There are several areas of future work and development for voice assistant technology. Some of the key areas include:

1. **Personalization and context-awareness**: Voice assistants are already able to recognize and adapt to individual users' speech patterns and preferences. However, there is potential to further improve personalization by incorporating more contextual information, such as a user's location, past interactions, and social media activity.

2. **Multimodal interfaces**: While voice assistants provide a hands-free and eyes-free interface, there is potential to incorporate additional modalities, such as touch and gesture-based interfaces. This could enable more flexible and intuitive interactions, especially in situations where voice commands may not be practical or appropriate.

3. **Integration with IoT devices**: Voice assistants are already widely used in smart home devices, but there is potential to expand their integration with other IoT devices, such as wearable devices and industrial sensors. This could enable more seamless and efficient interactions between users and their

devices.

4. **Natural language understanding and generation**: Voice assistants are already able to understand and generate a wide range of voice commands and responses, but there is potential to improve their natural language processing capabilities. This could include more advanced language models and machine learning algorithms, as well as the incorporation of semantic and contextual information.

5. **Improved security and privacy**: As voice assistants become more widely used and incorporate more personal and sensitive information, it is important to continue improving their security and privacy features. This could include more advanced encryption and authentication methods, as well as more transparent data management practices.

6. **Cross-platform integration**: With the proliferation of voice assistants from different providers, there is potential to improve cross-platform integration, enabling users to seamlessly switch between different voice assistants and devices. This could require standardization of voice command syntax and protocols.

Future work in voice assistant technology is likely to focus on improving their accuracy, efficiency, and personalization, while ensuring robust security and privacy features. There is also potential for voice assistants to become even more ubiquitous and integrated with other devices and services, transforming the way we interact with technology and the world around us.

7.3 RESEARCH ISSUES

There are several research issues in voice assistant technology that are currently being explored by researchers and developers. Some of the key research issues include:

Robustness and reliability: While voice assistants have improved significantly in recent years, they are still prone to errors and inaccuracies, especially in noisy or complex environments. Researchers are exploring new approaches to improve the robustness and reliability of voice assistants, such as using more advanced machine learning algorithms and incorporating contextual and semantic information.

Privacy and security: As voice assistants become more ubiquitous and incorporate more personal and sensitive information, there is a growing need to ensure their privacy and security features are robust and effective. Researchers are exploring new methods to improve the privacy and security of voice assistants, such as using more advanced encryption and authentication methods, and developing more transparent data management practices.

Multimodal interfaces: While voice assistants provide a hands-free and eyes-free interface, there is potential to incorporate additional modalities, such as touch and gesture-based interfaces. Researchers are exploring new approaches to develop multimodal interfaces that are more intuitive and efficient.

Natural language understanding and generation: While voice assistants are already able to understand and generate a wide range of voice commands and responses, there is potential to improve their natural language processing capabilities. Researchers are exploring new approaches to develop more advanced language models and machine learning algorithms that can better understand and

generate natural language.

Personalization and context-awareness: Voice assistants are already able to recognize and adapt to individual users' speech patterns and preferences, but there is potential to further improve personalization by incorporating more contextual information, such as a user's location, past interactions, and social media activity. Researchers are exploring new approaches to develop more advanced personalization and context-awareness features.

Ethical considerations: As voice assistants become more ubiquitous and integrated with our daily lives, there is a growing need to consider their ethical implications. Researchers are exploring the ethical considerations of voice assistant technology, such as issues around bias, transparency, and accountability.

Research in voice assistant technology is focused on improving their accuracy, efficiency, and personalization, while ensuring robust security and privacy features. There is also a growing focus on ethical considerations and the broader societal implications of voice assistant technology.

7.4 IMPLEMENTATION ISSUES

Here are some implementation issues that developers and organizations may need to consider when building and deploying voice assistant applications:

1. **Multilingual support**: Voice assistant applications may need to support multiple languages and dialects, depending on their target user base. Developers need to ensure that their applications can accurately recognize and respond to a wide range of language variations, and that they can be easily localized for different markets.

2. **Hardware requirements**: Voice assistant applications may require specific hardware requirements, such as microphones, speakers, and other audio

equipment. Developers need to carefully consider the hardware requirements of their applications, and ensure that they are compatible with the devices on which they will be used.

3. **Performance optimization**: Voice assistant applications may require optimization for performance, especially when dealing with large datasets or complex processing algorithms. Developers need to ensure that their applications are optimized for speed and efficiency, and that they can handle large volumes of user requests and interactions.

4. **Data management**: Voice assistant applications may generate large amounts of data, such as user preferences, voice recordings, and other user interactions. Developers need to ensure that they have effective data management processes in place, and that they are compliant with data protection regulations and industry standards.

5. **User feedback and analytics**: Voice assistant applications can generate valuable user feedback and analytics data, which can be used to improve the performance and effectiveness of the application over time. Developers need to ensure that they have effective user feedback and analytics processes in place, and that they are able to make data-driven improvements to their applications based on this feedback.

6. **Regulatory compliance**: Voice assistant applications may be subject to regulatory compliance requirements, such as privacy regulations, accessibility standards, or industry-specific guidelines. Developers need to ensure that their applications are compliant with all relevant regulations and standards, and that they can provide evidence of compliance if required.

7. **Maintenance and updates**: Voice assistant applications require ongoing maintenance and updates to ensure that they remain effective and up-to-date with the latest technology and user requirements. Developers need to

implement robust maintenance and update processes, and ensure that their applications can be easily updated and maintained over time.

8. **Maintenance and updates**: Voice assistant applications require ongoing maintenance and updates to ensure that they remain effective and up-to-date with the latest technology and user requirements. Developers need to implement robust maintenance and update processes, and ensure that their applications can be easily updated and maintained over time.

Overall, implementing a voice assistant application requires careful consideration of a wide range of technical, user, and regulatory requirements. Developers and organizations need to carefully plan and execute their voice assistant projects to ensure that they are effective, efficient, and compliant with all relevant regulations and standards.

REFERENCES

[1] Klüwer, Tina. "From chatbots to dialog systems." Conversational agents and natural language interaction: Techniques and Effective Practices. IGI Global, 2011. 1–22.

[2] Daniel B. Kline (30 January 2017). "Alexa, How Big Is Amazon's Echo?". The Motley Fool.

[3] Markowitz, Judith. "Toys That Have a Voice". SpeechTechMag.

[4] Moskvitch, Katia. "The machines that learned to listen". BBC. Retrieved 5 May 2020.

[5] Epstein, J; Klinkenberg, W. D (1 May 2001). "From Eliza to Internet: a brief history of computerized assessment". Computers in Human Behavior. 17 (3):295–314. doi:10.1016/S0747-5632(01)00004-8. ISSN 0747-5632.

[6] Weizenbaum, Joseph (1976). Computer power and human reason : from judgment to calculation. Oliver Wendell Holmes Library Phillips Academy. San Francisco : W. H. Freeman.

[7] "Smartphone: your new personal assistant – Orange Pop". 10 July 2017. Archived from the original on 10 July 2017. Retrieved 5 May 2020.

[8] Darren Murph (4 October 2011). "iPhone 4S hands-on!". Engadget.com. Retrieved 10 December 2017.

[9] "Feature: Von IBM Shoebox bis Siri: 50 Jahre Spracherkennung – WELT" [From IBM Shoebox to Siri: 50 years of speech recognition]. Die Welt (in German). Welt.de. 20 April 2012. Retrieved 10 December 2017.

[10] "Conversica Raises $31 Million in Series C Funding to Fuel Expansion of Conversational AI for Business". Bloomberg.com. 30 October 2018. Retrieved 23 October 2020.

APPENDIX

A. SOURCE CODE

```
engine = pyttsx3.init("sapi5")
voices = engine.getProperty("voices")
engine.setProperty("voice", voices[0].id)
rate = engine.setProperty("rate",170)

def speak(audio):
    engine.say(audio)
    engine.runAndWait()

def takeCommand():
    r = speech_recognition.Recognizer()
    with speech_recognition.Microphone() as source:
        print("Listening.....")
        r.pause_threshold = 1
        r.energy_threshold = 300
        audio = r.listen(source,0,4)

    try:
        print("Understanding..")
        query  = r.recognize_google(audio,language='en-in')
        print(f"You Said: {query}\n")
    except Exception as e:
        print("Say that again")
        return "None"
    return query
```

Fig – 6.1 speech recognition

The above code represents speech recognition module which tends to take input, provide the output of the audio and recognize it in the preferred language. The program helps to analyze the voice of an user and provide the speech to text. Speech rate, output and enrichened in this stage. By applying this to all the functions or features in our model it will be a handy.

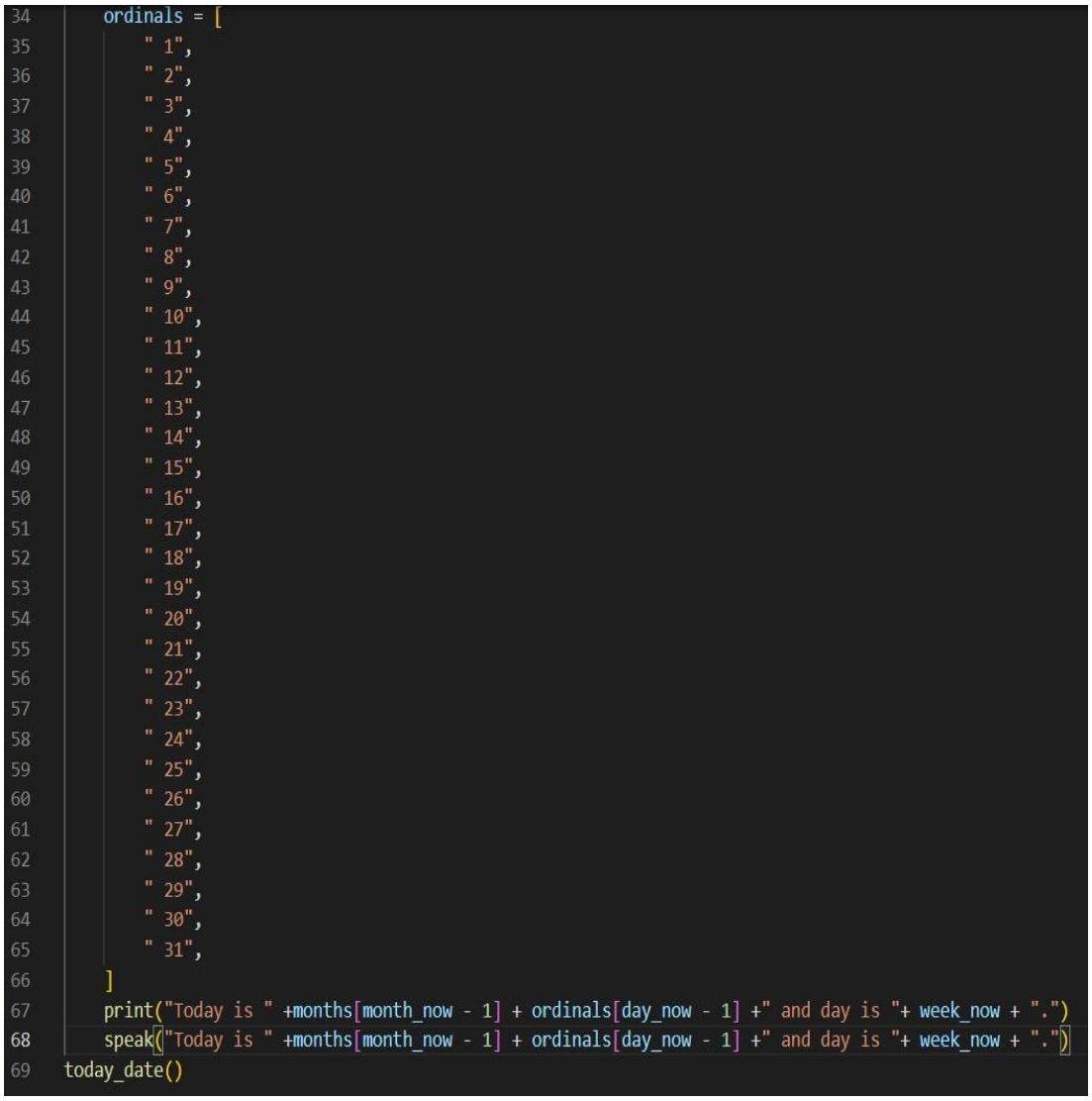
```
extractedtime = open("Alarmtext.txt","rt")
time = extractedtime.read()
Time = str(time)
extractedtime.close()

deletetime = open("Alarmtext.txt","r+")
deletetime.truncate(0)
deletetime.close()

def ring(time):
    timeset = str(time)
    timenow = timeset.replace("jarvis","")
    timenow = timenow.replace("set an alarm","")
    formatted_timenow = "{}:{}:{}".format(timenow[0:2], timenow[2:4], timenow[4:])
    Alarmtime = str(formatted_timenow)
    print(Alarmtime)
    while True:
        currenttime = datetime.datetime.now().strftime("%H:%M:%S")

        if currenttime == Alarmtime:
            speak("Alarm ringing,sir")
            os.startfile(r'D:\ex1\music\heat waves.mp3')
```

Fig – 6.2 alarm

Alarm function represents in fig 6.2. to set the alarm user must be confident in away of sating the 24 hr format with seconds in numerical format for example if the person wants to create alarm at 5:40 pm. Then the time must be spell as 174001. Here 01 represents seconds. So the user must be aware of these to ring the alarm.

```
def today_date():
    now = datetime.datetime.now()
    date_now = datetime.datetime.today()
    week_now = calendar.day_name[date_now.weekday()]
    month_now = now.month
    day_now = now.day
    months = [
        "January",
        "February",
        "March",
        "April",
        "May",
        "June",
        "July",
        "August",
        "September",
        "October",
        "November",
        "December",
    ]
```

Fig 6.3 month



```
    ordinals = [
        " 1",
        " 2",
        " 3",
        " 4",
        " 5",
        " 6",
        " 7",
        " 8",
        " 9",
        " 10",
        " 11",
        " 12",
        " 13",
        " 14",
        " 15",
        " 16",
        " 17",
        " 18",
        " 19",
        " 20",
        " 21",
        " 22",
        " 23",
        " 24",
        " 25",
        " 26",
        " 27",
        " 28",
        " 29",
        " 30",
        " 31",
    ]
    print("Today is " +months[month_now - 1] + ordinals[day_now - 1] +" and day is "+ week_now + ".")
    speak("Today is " +months[month_now - 1] + ordinals[day_now - 1] +" and day is "+ week_now + ".")
today_date()
```

Fig – 6.4 date and day

Fig 6.3 and 6.4 are combined functions. Both results single opeation. By applying this query we can get what day, month, date is. Query's output will be predicted in the region where the system location is allocated.

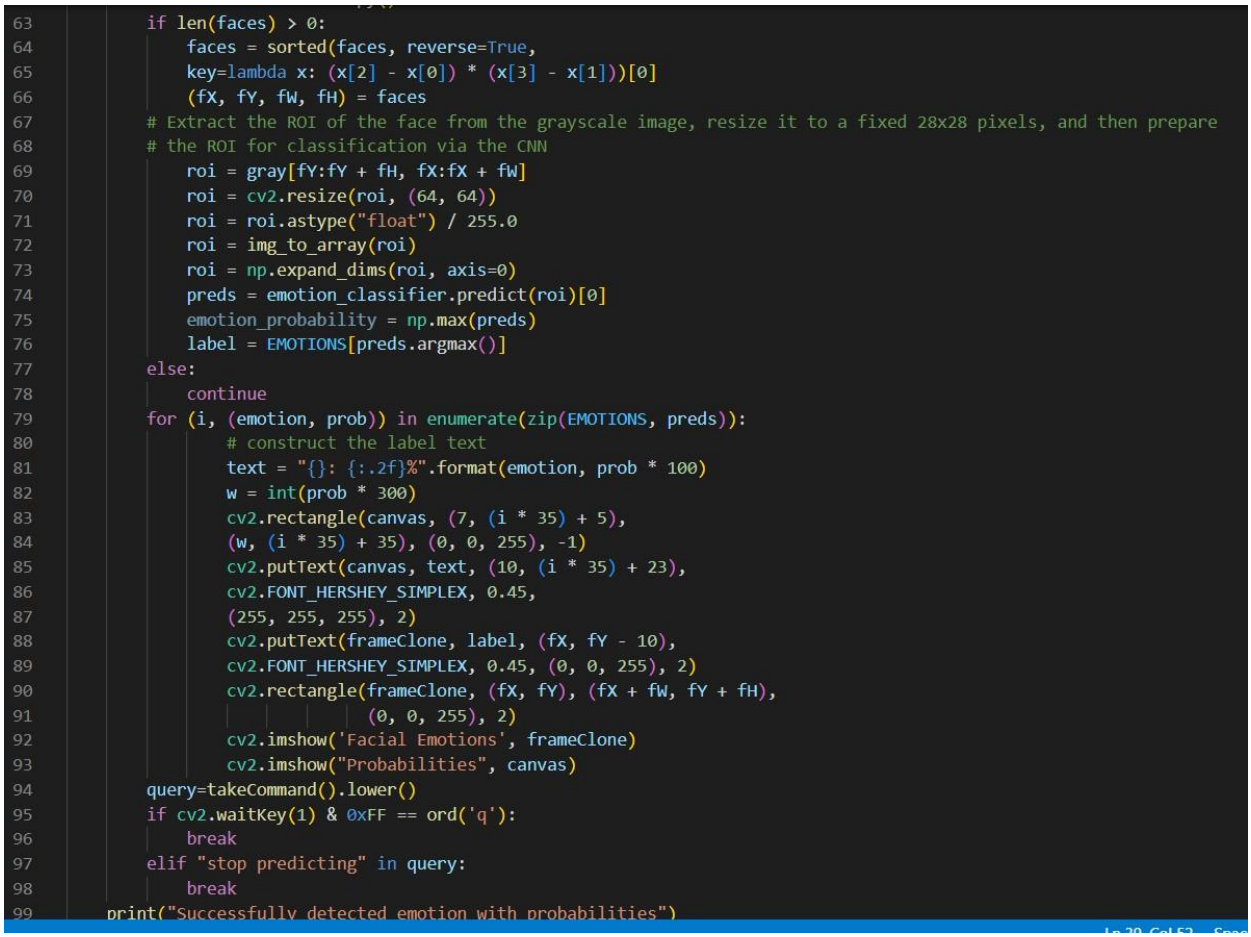
```
def openappweb(query):
    speak("Launching, sir")
    if ".com" in query or ".co.in" in query or ".org" in query or ".in" in query:
        query = query.replace("open","")
        query = query.replace("jarvis","")
        query = query.replace("launch","")
        query = query.replace(" ","")
        webbrowser.open(f"https://www.{query}")
    else:
        keys = list(dictapp.keys())
        for app in keys:
            if app in query:
                os.system(f"start {dictapp[app]}")

def closeappweb(query):
    speak("Closing,sir")
    if "one tab" in query or "1 tab" in query:
        pyautogui.hotkey("ctrl","w")
        speak("one tab closed")
    elif "2 tab" in query:
        pyautogui.hotkey("ctrl","w")
        sleep(0.5)
        pyautogui.hotkey("ctrl","w")
        speak("two tabs closed")
    elif "3 tab" in query:
```

Fig – 6.5 open/close apps

If the application is in system and user make it application available access to the program then the application will be opened according to the query given or else if the query ends with .com, .in, .org, .co.in then chrome browser will be opened and searches accordingly. This is what fig 6.5 openappweb represents. Similarly for closing the tab or application we can use respective close functionality given to it.

```
emotion_model_path = '_mini_XCEPTION.102-0.66.hdf5'
# hyper-parameters for bounding boxes shape
# loading models
face_cascade=cv2.CascadeClassifier(cv2.data.haarcascades + "haarcascade_frontalface_default.xml")
emotion_classifier = load_model(emotion_model_path, compile=False)
EMOTIONS = ["angry" ,"disgust","scared", "happy", "sad", "surprised",
 "neutral"]
camera = cv2.VideoCapture(0,cv2.CAP_MSMF)
camera.set(cv2.CAP_PROP_FRAME_HEIGHT, 480)
camera.set(cv2.CAP_PROP_FRAME_WIDTH, 640)
camera.set(cv2.CAP_PROP_FOURCC, 0x32595559)
camera.set(cv2.CAP_PROP_FPS, 25)
def emotions():
    # starting video streaming
    cv2.namedWindow('Facial Emotions')
    camera = cv2.VideoCapture(0,cv2.CAP_DSHOW)
    while True:
        frame = camera.read()[1]
        #reading the frame
        frame = imutils.resize(frame,width=600)
        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
        faces = face_cascade.detectMultiScale(gray,scaleFactor=1.1,minNeighbors=5,minSize=(30,30),flags=cv2.CASCADE_SCALE_IMAGE)
        canvas = np.zeros((250, 300, 3), dtype="uint8")
        frameClone = frame.copy()
```

Fig – 6.6 emotion recognition part1



```
        if len(faces) > 0:
            faces = sorted(faces, reverse=True,
            key=lambda x: (x[2] - x[0]) * (x[3] - x[1]))[0]
            (fX, fY, fW, fH) = faces
        # Extract the ROI of the face from the grayscale image, resize it to a fixed 28x28 pixels, and then prepare
        # the ROI for classification via the CNN
            roi = gray[fY:fY + fH, fX:fX + fW]
            roi = cv2.resize(roi, (64, 64))
            roi = roi.astype("float") / 255.0
            roi = img_to_array(roi)
            roi = np.expand_dims(roi, axis=0)
            preds = emotion_classifier.predict(roi)[0]
            emotion_probability = np.max(preds)
            label = EMOTIONS[preds.argmax()]
        else:
            continue
        for (i, (emotion, prob)) in enumerate(zip(EMOTIONS, preds)):
                # construct the label text
                text = "{}: {:.2f}%".format(emotion, prob * 100)
                w = int(prob * 300)
                cv2.rectangle(canvas, (7, (i * 35) + 5),
                (w, (i * 35) + 35), (0, 0, 255), -1)
                cv2.putText(canvas, text, (10, (i * 35) + 23),
                cv2.FONT_HERSHEY_SIMPLEX, 0.45,
                (255, 255, 255), 2)
                cv2.putText(frameClone, label, (fX, fY - 10),
                cv2.FONT_HERSHEY_SIMPLEX, 0.45, (0, 0, 255), 2)
                cv2.rectangle(frameClone, (fX, fY), (fX + fW, fY + fH),
                              (0, 0, 255), 2)
                cv2.imshow('Facial Emotions', frameClone)
                cv2.imshow("Probabilities", canvas)
        query=takeCommand().lower()
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
        elif "stop predicting" in query:
            break
    print("Successfully detected emotion with probabilities")
```

Fig – 6.7 emotion recognition part 2

Fig 6.6 and 6.7 both resembles emotion recognition in fig 6.6 we have a hdf5 file that have the trained and validated images with labelled emotions and haarcascade file for facial values deployment setup with hdf5 file and the required emotions are highlighted. Using the webcam, the emotions are captured with the facial expression. There will be 2 gui screens one shows the live image and shows the probabilities of emotions produced.

```
def calculator(query):
    if "calculate" in query:
        try:
            client = wolframalpha.Client(app_idapikey)
            ind = query.lower().split().index("calculate")
            query = query.split()[ind + 1:]
            res = client.query(" ".join(query))
            answer = next(res.results).text
            print("The answer is " + answer)
            speak("The answer is " + answer)
        except:
            print("not available")
            speak("please check the values")

def gk(query):
    if "what is" or "who is" in query:
        try:
            client = wolframalpha.Client(app_idapikey)
            ind = query.lower().split().index("is")
            query = query.split()[ind + 1:]
            res = client.query(" ".join(query))
            answer = next(res.results).text
            print(answer)
            speak(answer)
        except:
            print("not available")
            speak("please check the statement")
```

Fig – 6.8 wolfram alpha

```
35  keyboard = Controller()
36
37  def volumeup():
38      for i in range(5):
39          keyboard.press(Key.media_volume_up)
40          keyboard.release(Key.media_volume_up)
41          sleep(0.1)
42
43  def volumedown():
44      for i in range(5):
45          keyboard.press(Key.media_volume_down)
46          keyboard.release(Key.media_volume_down)
47          sleep(0.1)
```

Fig – 6.9 keyboard controls

The keyboard controls are meant to be controlled with the keyboard but here we use these controls via suitable voice. For example, to turn volume up and down there is simple command that perform this task as shown in fig 6.9.

```
def whatsapp():
    speak("Tell me person name or number or group.")
    person_name = takeCommand().lower()
    speak("say the message you want me to send.")
    my_msg = takeCommand().lower()
    webbrowser.open('https://web.whatsapp.com/')
    time.sleep(12)
    print(pyautogui.position())
# click on search bar
    pyautogui.click(132,257)
    pyautogui.typewrite(person_name)
    time.sleep(5)
#click on person
    pyautogui.click(163,319)
    time.sleep(5)
    pyautogui.typewrite(my_msg)
    time.sleep(2)
    pyautogui.click(1846,951)
```

Fig – 7.0 whatsapp msg

Sending a whatsapp message using chrome browser as supportive platform is explained in fig 5.0. here we use the screen gui and with just a simple command to let know of whom to send and what to send the system will directly send the message to the respective query made.

```
def searchGoogle(query):
    if "google" in query:
        import wikipedia as googleScrap
        query = query.replace("google search","")
        query = query.replace("Google","")
        query = query.replace("google","")
        pywhatkit.search(query)
        speak("This is about "+query+" on google")

def searchYoutube(query):
    if "youtube" in query:
        query = query.replace("youtube search","")
        query = query.replace("youtube","")
        web = "https://www.youtube.com/results?search_query=" + query
        webbrowser.open(web)
        speak("This is about"+query+" on youtube")

def searchWikipedia(query):
    if "wikipedia" in query:
        speak("Searching from wikipedia....")
        query = query.replace("wikipedia","")
        query = query.replace("search wikipedia","")
        try:
            Results = wikipedia.summary(query,sentences = 2)
            speak("According to wikipedia..")
            print(Results)
            speak(Results)
        except:
            speak("no output available on "+query)
```

Fig – 7.1 Query searches

```
def latestnews():
    api_dict = {"business" :"https://newsapi.org/v2/top-headlines?country=in&category=business&apiKey="+newsapikey ,
            "entertainment" : "https://newsapi.org/v2/top-headlines?country=in&category=entertainment&apiKey="+newsapikey,
            "health" : "https://newsapi.org/v2/top-headlines?country=in&category=health&apiKey="+newsapikey,
            "science" :"https://newsapi.org/v2/top-headlines?country=in&category=science&apiKey="+newsapikey,
            "sports" :"https://newsapi.org/v2/top-headlines?country=in&category=sports&apiKey="+newsapikey,
            "technology" :"https://newsapi.org/v2/top-headlines?country=in&category=technology&apiKey="+newsapikey}

    url = None
    speak("Which field news do you want, [business] , [health] , [technology], [sports] , [entertainment] , [science]")
    field = takeCommand().lower()
    print(field)
    for key ,value in api_dict.items():
        if key.lower() in field.lower():
            url = value
            print(url)
            print("url was found")
            break
        else:
            url = True
    if url is True:
        print("url not found")

    news = requests.get(url).text
    news = json.loads(news)
    speak("Here is the first news.")

    arts = news["articles"]
    for articles in arts :
        article = articles["title"]
        print(article)
        speak(article)
        news_url = articles["url"]
        print(f"for more info visit: {news_url}")
        speak("To continue say yes. To stop the news say close")
        field = takeCommand().lower()
        if str(field) == "yes":
```

Fig -7.2 news

News functionality. there are different categories of news and according to user choice the required categorical news can be selected. And based on that category the news will b provided. The news will be provided continuously unless user mention to a voice command “stop” to close the news.

```
def reminder(query):
    if "remember that" in query:
        remindermessage = query.replace("remember that","")
        speak("You told me to remember that"+remindermessage)
        reminder = open("reminder.txt","a")
        reminder.write(remindermessage)
        reminder.close()
    elif "what do you remember" in query:
        reminder = open("reminder.txt","r")
        speak("Reminders are" + reminder.read())
    elif "delete all reminders" in query:
        delete = open("reminder.txt","r+")
        delete.truncate(0)
        delete.close()
```

Fig – 7.3 reminders

```python
def wallpaper(query):
    if "change my wallpaper" in query:
        img = r'D:\ex1\wallpaper'
        list_img = os.listdir(img)
        imgChoice = random.choice(list_img)
        randomImg = os.path.join(img, imgChoice)
        ctypes.windll.user32.SystemParametersInfoW(20, 0, randomImg, 0)
        speak("Background changed successfully")

def music(query):
    if "play song" in query:
        speak("Here you go with music")
        music_dir = r'D:\ex1\music'
        songs = os.listdir(music_dir)
        random.shuffle(songs)
        os.startfile(os.path.join(music_dir,songs[0]))
    elif "close song" in query:
        os.system('taskkill /f /im Microsoft.Media.Player.exe')


def recyclebin(query):
    if "recycle bin" in query:
        winshell.recycle_bin().empty(
            confirm=True, show_progress=False, sound=True
        )
        speak("Recycle Bin Emptied")
```

Fig – 7.4 wallpaper, music, recyclebin

```python
def currentweather(query):
    if "weather" in query:
        weather_url = "http://api.openweathermap.org/data/2.5/weather?"
        ind = query.split().index("in")
        location = query.split()[ind + 1:]
        location = "".join(location)
        url = weather_url + "appid=" + weatherapikey + "&q=" + location
        js = requests.get(url).json()
        try:
            if js["cod"] != "404":
                weather = js["main"]
                temperature = weather["temp"]
                temperature = temperature - 273.15
                humidity = weather["humidity"]
                desc = js["weather"][0]["description"]
                weatherResponse = " The temperature in Celcius is " + str(temperature)
                print(weatherResponse)
                speak(weatherResponse)

        except:
            speak("City Not Found")
```

Fig – 7.5 weather report

```
def greetMe():
    hour  = int(datetime.datetime.now().hour)
    if hour>=6 and hour<=12:
        speak("Good Morning sir")
        print("Good Morning sir")
    elif hour >12 and hour<=18:
        speak("Good Afternoon sir")
        print("Good Afternoon sir")
    elif hour >18 and hour<=20:
        speak("Good Evening sir")
        print("Good Evening sir")
    elif hour >20 and hour <6:
        speak("Sir, Its time for you to go and get some sleep.")
        print("Sir, Its time for you to go and get some sleep.")
    speak("Please tell me, How can I help you ?")
greetMe()
```

Fig – 7.6 wake wishes

B. SCREENSHOTS

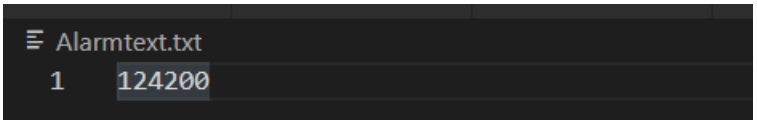


Fig – 8.1 alarm output

Fig – 8.2 set voices

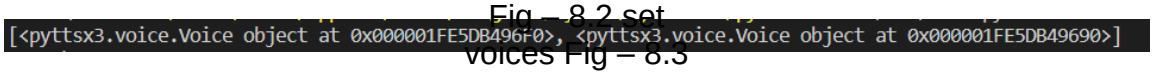


Fig – 8.3 day date

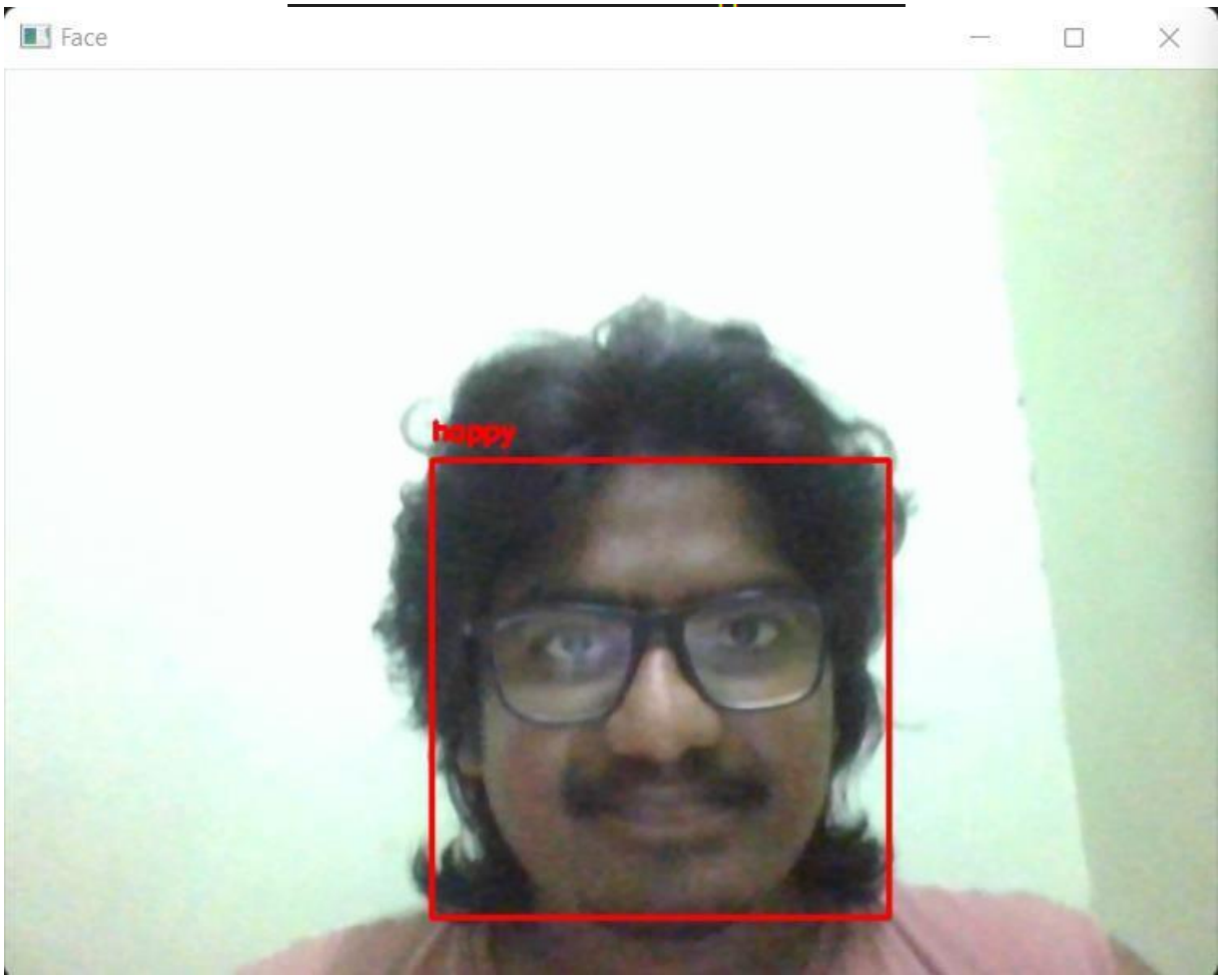


Fig – 8.4 facial emotion

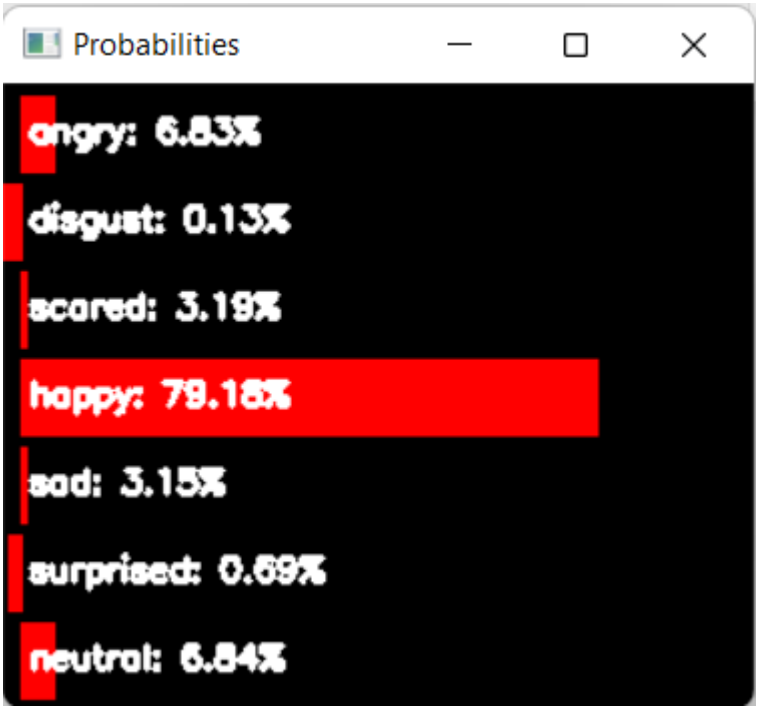


Fig – 8.5 emotions probability

```
Listening....
Understanding....
result2:
{   'alternative': [   {   'confidence': 0.88687539,
                           'transcript': 'calculate the √16'}],
    'final': True}
You Said: calculate the √16

The answer is 4
```

Fig – 8.6 calculator

```
Listening....
Understanding....
result2:
{   'alternative': [   {   'confidence': 0.88687533,
                           'transcript': 'who is the president of India'}],
    'final': True}
You Said: who is the president of India

Draupadi Murmu (from 25/07/2022 to present)
```

Fig – 8.7 GK wolframalpha

Fig – 8.8 news

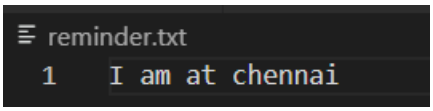


Fig – 8.9 reminders

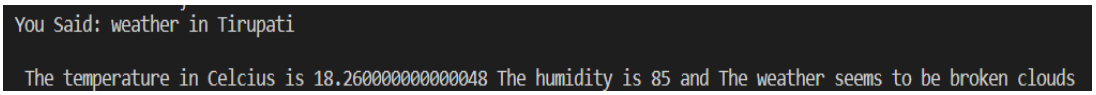


Fig – 9.0 weather report

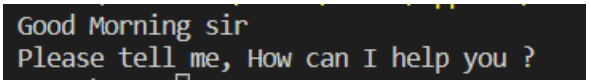


Fig – 9.1 greetings

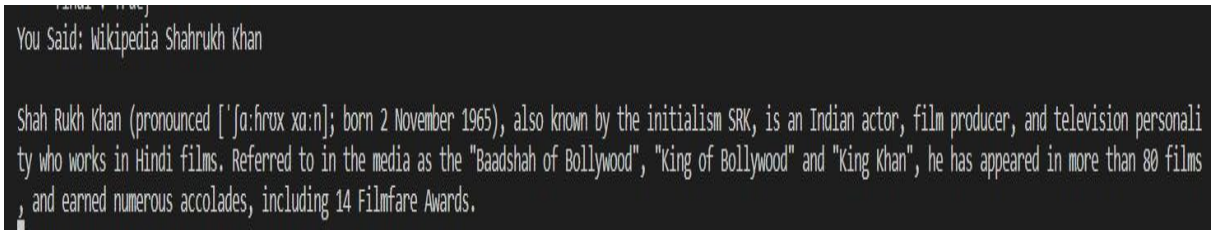


Fig – 9.2 Wikipedia search

C.RESEARCH PAPER

VOICE ASSISTANT WITH EMOTION RECOGNITION

S. Sravana Srikanth,
Student
Department of Computer Science And Engineering
Sathyabama Institute of Science And Technology.
Chennai, India.
sunkararsikanth002@gmail.com

Sreenivasan Manikantan,
Student
Department of Computer Science And Engineering
Sathyabama Institute of Science And Technology.
Chennai, India.
Seenumani141@gmail.com

A. Mary Posonia,
Associate Professor
Department of Computer Science And Engineering
Sathyabama Institute of Science And Technology.
Chennai, India.
maryposonia.cse@sathyabama.ac.in

***Abstract*--** **Voice assistants have become a essential part in our daily life, which helps to provide essential tasks in a hands free manner. Here the voice assistant helps the user with hands free experience that includes setting reminders, playing music, alarm, wishes etc. The voice assistant is powerful ones of cutting-edge speech recognition and natural language processing technique. The system will be intended to interpret and respond to a wide range of spoken instructions and enquiries in a natural and intuitive manner. Furthermore, the initiative intends to improve the user experience by giving tailored replies and suggestions depending on the user's preferences and history. For forecasting emotions, features such as emotion recognition, which detects a person's feelings through their face, are implemented. The goal of this project is to create a voice assistant that interacts with and shorten the burden of activities.**

***Keywords*—voice assistant, api key, emotion recognition, messages, alarm, news report**.

I. INTRODUCTION

Voice assistants are AI developed cutting-edge voice assistant that will assist users in various tasks such as setting reminders, sending a message to user via communication networks and help to reduce the cost of effort of typing or contacting, playing music etc. It will be focusing on improving the user experience by providing personalized responses and suggestions based on the user's preferences, suggestions, needs, making the voice assistant a valuable tool in user's daily life. The goal of this project is to create a voice assistant that intuitive to its relevance and easy to use, providing a seamless experience.

Although Major assistants like google, Siri, Alexa are growing rapidly, they do not provide the enough support for working environment, lack of sustainability etc. this can be replaced through a virtual assistant which helps the user getting things done and provide seamless experience.

II. LITERATURE SURVEY

Wenrui Diaoinherit [1] work shows difference in From a long time, the research based on sensor attacks only focused on primarily accessing or controlling the main parts such as camera microphone these parts directly get the required permission from the sensors this paper discusses about the best way to bypass these android apps without any permission. By using technology of voice search through a speaker. The android intent mechanism, VOIC employer initiates operations, they can send sms / email access privacy information transmit sensitive data archives remote control. It also detects glitch in google search app which is utilized to dial some numbers even when phone is not active. This study helps us to know that even sound devices like speakers can also be used to breach our privacy protocol.

Lewel soni, [2] inherit work shows difference in Voice assistant helps us to operate mobile devices by communicating our inputs as commands and hearing the output but this wonderful technology can sometimes be hacked and used for wrong purposes like liking of personal information etc. This paper discusses about a mechanism which helps to keep the voice assistant safe from these kind of privacy inclusions. This technology can detect the voice and if it feels that it not secures to execute the command it will give warning sign and ask for the authentication of the user who is using it.

Huan feng [3] inherit work shows difference in Google voice search takes the input by understanding the user commands executing it and giving the output. Whenever so this paper focuses mainly on vulnerabilities and how this technology can be used for hacking the systems and can be

used to steal personal sensitive information. These kinds of papers can be used to detect the vulnerabilities and resolve them.

Emi moriuchi [4] work shows difference in Communication helps us to let our feelings and to get our work done by others, nowadays communication is also important due to the presence of technologies like personal voice assistant which basically convert the spoken words into a command and give the require output in spoken form. So, this paper mainly focuses on robot which acts on the spoken command of the user and gives the require output by speaking out. It was set of inbuilt datasets and make use of technologies like natural language processing to understand the command the chip present inside the robot will help it execute the command understood and give the respective output. This kind of technologies is very useful for the people with disabilities as some kind of daily activities which are performed by fully fit humans cannot be performed as proficiently by these peoples so these robots by combinations of sensors and actuators network will help them in every walk of their life.

Alisha Pradhan [5] inherit work shows difference in The digital twin is an evolving concept with many applications which are being recently discovered actually voice assistant like google, Alexa, Siri are used are execute to understand and execute the commands given by the user but this paper focuses on taking this technology to next level. By using this technology, the quality of the product output can be predicted which are made in factories so this paper mainly focuses on technology by which a voice assistant can be trained by the use no dataset to predict the defect of the produce of a particular production line and also help us solve the problem of low quality at in at economically.

Greame mclean [6] inherit shows difference in Artificially intelligence and voice control gadgets have gained huge popularity over the last few years in the sector of home appliances. On further analysis this paper aim that predicting the region for rapid raise of use of these appliances. They are basically three-line reasons by which it can be concluded, why these devises have gained such popularity over the last few years. 1. Utility benefits, 2. Symbolic benefits, 3. Social benefits. Because they are voice assistant people are getting attracted towards this technology as humans want to live very comfortable life. The unique functionalities of these gadgets is dragging the user base towards itself in today's market.

III. PROPOSED METHODOLOGY

In this paper proposed work is to identify various tasks are implemented using our assistant and retract through it. With the usage of voice commands things done easier and efficient way through coded AI assistant. Performing tasks with voice commands and improving it further for n number of tasks that are needed to attain and move forward through technologies. Innovations can be dealt in this way. Before knowing about this the assistant must be woken up with the wake-up function, thereby preclaims an intermediate cause or disturbance does not provide chunk or clumsy to assistant. Tasks are needed to define with its given time range. Excluding time range does not produce anything and provide negligible data. Although time range can be set manageable, it can't get the point or a conclusion about hour speech or vast range of query. The flow chart is attached in the given paper for better understanding of the workflow and better to understand.

Natural Language Processing (NLP) and speech recognition module helps to obtain the best outcome through labelled input voice data. By obtaining the audio from various user the nlp is trained and brought with its speech recognition module to perform the audio processing in any harsh condition and there by proceeding with necessary needs.

Emotion Recognition is to identify possessive emotion from a person and evaluate the need accordingly. It restrains the mood of approach and provide essential output required to balance the emotion by providing multiple tasks etc.

IV. EXPERIMENTAL SETUP

HDF5-Hierarchical Data Format also represented as h5 file. Load the h5 file for processing emotions. The h5 file consists of attributes that contains images stored in the form of raw data in the statistical method and additional sensors result in detecting the image and its sources. These are in form of metadata containing different files that synchronize the images with available sensors values and can easily track this context information thereby predicts the live emotion, respectively. In this manner the h5 file can be accessed in multiple views with n number of ways and by necessary functions that are needed can be retrieved directly, organize data. Note that h5 are not human readable.

Xml file—extensible markup language for processing the hdf5 data. In XML use <> tags or marks. Although it is similar to html, the tags are

not predefined, not limited and it can be extended

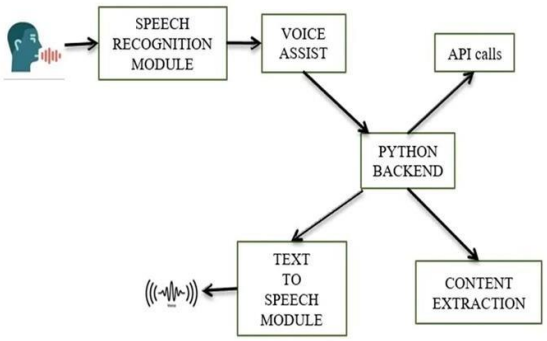
when compared with html. It refers for data exchange between application. It does not depend upon any platform. It is both human and machine- readable file.

Create a python file and store the api key w.r.t the use case, procedure and initialize with a variable for identifying.

Functions implemented in proposed system are labelled in below fig 1.1. This represents the complete overflow of the voice assistant being processed in background and reminding the status.

Fig – 1.1

System Architecture



Below are set of major function's that are implemented and given below in a tabular format. Table- 1.1. This table represents how the output is evaluated for specific query and result mentions it.

Table – 1.1

Query & Output

Query	Function/Module	Result
news	Apikey+ news	News based on different sectors.
Alarm	datetime	Rings alarm
Today's date, time	datetime	Respective date time at that region
Search engine	Google, Wikipedia, you tube	Produce output with through provided search engine
weather	Apikey+ location	Provide weather report for given query about a place.
close	os	applicationin OS.
Change wallpaper	Ctypes, os	Replace currentwallpaper from aspecifiedfolder ofimages.
Recycle bin	winshell	Emptyitemsin therecyclebin.
songs	Startfile,os, taskkill	Canplay or stopthesong.
volume	Pynput.keyboard	Increase/decrease
"Based on provided input from a query "	wolframalpha+ Apikey+ query	Produces output for mathematical calculation, complex problems in science, GK etc.
Open/	Pyautogui,	Open/close

From the given data after calculating with 100

epochs, we got the accuracy of trained set is 94.1, validation set is 61.4 in accuracy graph. and the loss of trained set is 0.12, validation set is 1.64 in loss graph. To decrease loss or validation, the solution is to increase epochs, number of layers, hidden units, CNN, validation steps in generator. One of the major drawbacks is this model requires best graphical processor unit processor to validate and need more duration for estimating.

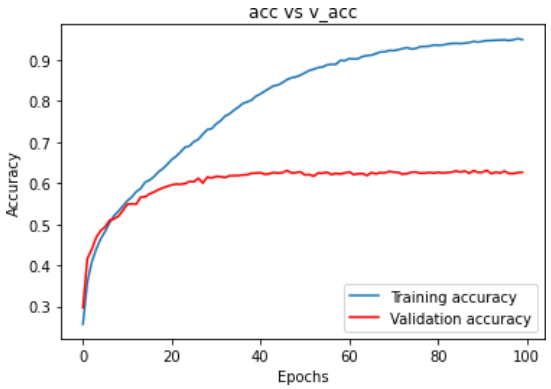
Table – 2.1

Accuracy vs Validation Accuracy

Epochs	Accuracy	Validation
20	0.6455	0.5919
40	0.8114	0.6244
60	0.8973	0.6246
80	0.9357	0.6242
100	0.9490	0.6263
	Result **94%**	

Fig – 2.1

Accuracy Graph



The figure 2.1, 2.2 shows about accuracy of dataset being trained and validating it (2.1), also the loss is predicted in 2.2. note that the epochs taken here are 100 and time took to neural network is nearly 4 hours (rough estimation, might varies). Table 2.1 represents convolutional neural network trained

model at different epochs by validating it with images the accuracy and loss is generated. At 100th epoch the accuracy is 94% and loss is 14%. Fig 2.2 shows Loss and validation loss.

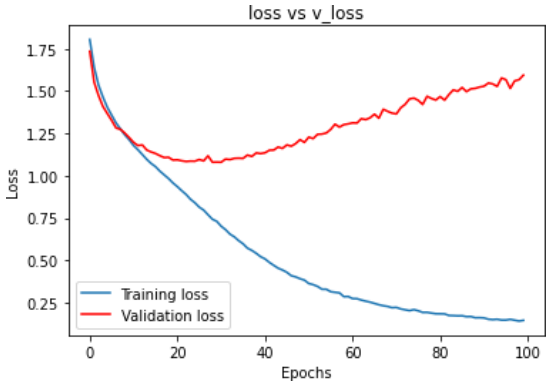
Table – 2.2

Loss vs Validation Loss

Epochs	Loss	Validation
20	0.9570	1.0924
40	0.5222	1.1308
60	0.2870	1.3062
80	0.1852	1.4467
100	0.1462	1.5940
	Result **14%**	

Fig – 2.2

Loss Graph



Voice assistant's emotion recognition model will recognize our mood at that interval of time by using a webcam allocated in that device and try to predict their emotion. For suppose if the person seems sad then the voice will ask the person a query like hey," you seem to be sad how can I help you to uplift you". Then if the person gives the rerun voice command it will do the same for the person or else it will be quiet after 4 seconds in this way this voice assistant acts as your friend during your down time and this feature can solve one of the biggest problems of human life in current scenario which is depression.

EMOTION RECOGNITION

In this section two data sets are engaged to train and validate. After the predictions are over, the data is stored in a h5 file for further emotion prediction and produce output. Provide xml that allocate with respective attribute to the h5 file which consists of prediction model and enhance with facial sectors and loaded data.

Below fig 3.1 represents the live update of person in fig 3.2 and probabilities of the emotions varies with every 0.2 th second taken at 20 fps.

Fig – 3.1

Emotions probability

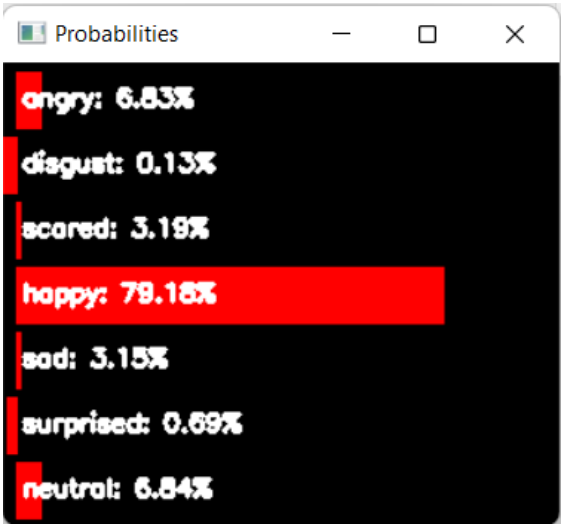
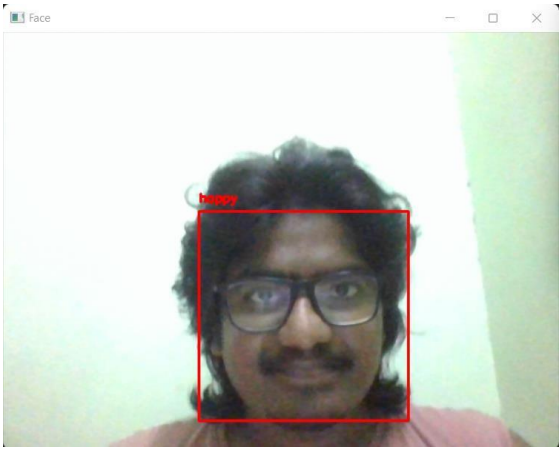


Fig – 3.2

Sample Image



Emotion recognition is a technology by which the system tries to know the mindset of the person who is using it. Emotion recognition is one of the key areas which this project tries to focus on by using this module the focus is to find the current mood of our client and to suggest things based on it to make him feel more satisfied with the working of our assistant the module used behind this is advanced image processing techniques which makes use of Convolutional Neural Network which is used to breakdown the image into various small pixels and

analyze these pixels for a better understanding of the mood.

Pass image through a convolutional neural network where the image is broken down further and every pixel of these images is analyzed after this, they are matched with the large number of images that are present in existing trained dataset then when it matches with image or a mixture of provided images based on number and the percentage of matching the predictions are engaged and results percentage of anger, happiness, sadness, neutral and a combination of many more moods base on person, the amount of accuracy of this prediction depends on the number of images that are trained in the dataset once all this output data about the person is out

V. CONCLUSION

At the end of this project I would like to conclude that AI is the future, and the voice assistants are the most basic part of AI in future. It would be engaged in sectors being taken over by these AI bots so that the load on the humans is reduced and the productivity of the activities in which people are involved in increases which might be unbeneficial for humankind.

Also because of these assistants there is no need to type something or even try opening something, just have to open our device and call out the voice assistant's name then it will do whatever the work that have on the internet. just have to provide the voice command of the required activity that has to be done and the assistant will take care of everything right from understanding our input to giving us the required output.

In this voice assistant there is this feature called emotion recognition which helps in recognizing the mood of the user and also giving out the proper suggestions based on that so for instance if someone is sad then it automatically recognizes that and suggests them things to enlighten their mood thereby can do a wholesome lot of things by using this like reading out news headlines giving weather updates and many more things which people do by typing out on internet without this assistant.

AI is a tool which is made to make the life of a human easy as the computer is doing things that a human can do as intelligently as a human so that the life of humans becomes very easy as machine can take care of most of the things as perfectly as humans so there is less amount of work that the humans have to consider so the productivity increases and the portion of the work people have to concentrate on is also reduced. But always have to be very careful about one thing in future that, make sure not to make them (refers to AI) so advance that it even might start control over mankind which eradicate everything and involves in every action.

VI. REFERENCES

[1] Wenrui Diao; Xiangyu Liu; Zhe Zhou; Kehuan Zhang; "Your Voice Assistant Is Mine: How To Abuse Speakers To Steal Information And Control Your Phone", ARXIV-CS.CR, 2014.

[2] Liwei Song; Prateek Mittal; "Inaudible Voice Commands", ARXIV-CS.CR, 2017.

[3] Matthew B Hoy; "Alexa, Siri, Cortana, And More: An Introduction To Voice Assistants", MEDICAL REFERENCE SERVICES QUARTERLY, 2018.

[4] Graeme McLean; Kofi Osei-Frimpong; "Hey Alexa ... Examine The Variables Influencing The Use of Artificial Intelligent In-home Voice Assistants", COMPUT. HUM. BEHAV., 2019.

[5] Tawfiq Ammari; Jofish Kaye; Janice Y. Tsai; Frank Bentley;

"Music, Search, and IoT", AC M TRANSACTIONS ON COMPUTER-HUMAN INTERACTION (TOCHI), 2019. (IF: 3)

[6] Emi Moriuchi; "Okay, Google!: An Empirical Study on Voice Assistants on Consumer Engagement and Loyalty", PSYCHOLOGY & MARKETING, 2019.

[7] Abhijit Guha; Timna Breßgott; Dhruv Grewal; **D.** Mahr; Martin Wetzels; Elisa B. Schweiger; "How Artificiality and Intelligence Affect Voice Assistant Evaluations", JOURNAL OF THE ACADEMY OF MARKETING SCIENCE, 2022.