A PROJECT REPORT

on

Speech Emotion Recognition Software

Submitted by

Ms. Rehmah Ahmed Batki

in partial fulfilment for the award of the degree of

BACHELOR OF SCIENCE

in

COMPUTER SCIENCE

under the guidance of

Assistant Prof. Rehana Ansari

Department of Computer Science

Since 1989
Devoted to Serve
Royal Higher Education Society's
ROYAL COLLEGE
OF ARTS, SCIENCE & COMMERCE
Empowerment through Value Education
(A Minority Institution Affiliated to University of Mumbai)

Royal College of Arts, Science and Commerce

Sem V

2023 – 2024

PROFORMA FOR THE APPROVAL PROJECT PROPOSAL

PNR No.**: ……** Roll no:**.....**

1. Name of the Student

_____

2. Title of the Project

_____

3. Name of the Guide :

_____

4. Teaching experience of the Guide _____

5. Is this your first submission? Yes ☐ No ☐

Signature of the Student Signature of the Guide

Date: …… Date: ……..

Signature of the Coordinator

Date: ……

Since 1989

Devoted to Serve

Royal Higher Education Society's

ROYAL COLLEGE
OF ARTS, SCIENCE & COMMERCE

Empowerment through Value Education

(A Minority Institution Affiliated to University of Mumbai)

Royal College of Arts, Science and Commerce,
BhaktiVedanta Swami Marg, Shrishti Complex,Mira Road East,
Maharashtra 401107.

Department of Computer Science

CERTIFICATE

This is to certify that Ms. Rehmah Ahmed Batki of **T.Y.B.Sc. (Sem V)** class has satisfactorily completed the Project Speech Emotion Recognition System, to be submitted in the partial fulfilment for the award of **Bachelor of Science** in **Computer Science** during the academic year **2023 – 2024.**

Date of Submission:

Project Guide

Head/Incharge,
Department Computer Science

College Seal

Signature of Examiner

DECLARATION

I, Rehmah Ahmed Batki, hereby declare that the project entitled "Speech Emotion Recognition Software" submitted in the partial fulfilment for the award of **Bachelor of Science** in **Computer Science** during the academic year **2023 – 2024** is my original work and the project has not formed the basis for the award of any degree, associateship, fellowship or any other similar titles.

Signature of the Student:

Place:

Date:

ACKNOWLEDGEMENT

Achievement is finding out what you would be doing rather than what you have to do. It is not until you undertake such a project that you realise how much effort and hard work it really is, what are your capabilities and how well you can present yourself or other things. It tells us how much we rely on the efforts and goodwill of others. It gives me immense pleasure to present this report towards the fulfilment of my project.

It has been rightly said that we are built on the shoulders of others. For everything I have achieved, the credit goes to all those who have helped me to complete this project successfully.

I take this opportunity to express my profound gratitude to **Founder Sir Prof. Asgar E. Lakdawala, Principal Dr. Kalpana Jain Patankar** and management of **Royal College of Arts, Science & Commerce, Mira Road (E)** for giving me this opportunity to accomplish this project work.

A special vote of thanks to **Prof. Ritika Lala (HOD) of Computer Science Department**, **Prof. Rehana Ansari** who is our professor & project guide for their most sincere, useful and encouraging contribution throughout the project span.

Finally, I would like to thank the entire Computer Science department who directly or indirectly helped me in completion of this project & to my family without whose support, motivation & encouragement this would not have been possible.

Rehmah Ahmed Batki

TABLE OF CONTENTS

List Of Figures

CHAPTER 1

INTRODUCTION

1. Introduction

Emotions help us to communicate with others, such as when we feel sad and need some help. They also can help us to act quickly in important situations. They can help you survive, grow, and connect with others. And they can guide your decisions, behaviours, and motivations. As babies, emotions are how you learn to communicate, even before you can talk.

In a world of cutting edge technology and its constant interaction with human beings, a seamless communication is a must to get desired response. While Natural Language processing (NLP) has made AI-Human interaction smooth, it leaves room for perception of unworded emotion.

Advances in machine learning and artificial intelligence offer promising solutions to bridge this gap by providing predictive models, guidance on preventive measures, and assistance in connecting with healthcare providers efficiently.

This project harnesses the power of machine learning, to predict emotions based on user's queries. Additionally, it offers a friendly chat-bot that is well equipped with NLP to provide perfect responses and solutions to any prompt.

Speech Emotion Recognition Software (SERS) is a smart AI tool that can converse via speech or text, respond well to any prompts and questions, perceive voice features and evaluate emotion behind one's voice.

1.1 Background and Project Overview

The project is purely python coded. The emotion detection system is based on the ML model shown in the following section. The model is trained on Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS). The dataset consists of 24 actors(12 Male, 12 Female) recordings of 8 emotions.

The model uses preprocessed dataset that is trained on 4 emotions that are Happy, Neutral, Sad and Angry. The model checks for features such as Chroma, Mel, MFCC (Mel-frequency cepstral coefficients) to train with the help of MLPClassifier (Multi-layer Perceptron classifier) which is a model that relies on an underlying Neural Network to perform the task of classification.

Prediction features are extracted with the help of python's Librosa library. Librosa is valuable Python music and sound investigation library that helps programming designers to fabricate applications for working with sound and music.

For Chatbot and NLP, the project uses Openai's API. The user prompts are sent to Openai servers and a fitting response is sent back to the software with the help of API key.

For profiles, login data and emotion history the software uses SQL database. The database stores usernames, passwords, profile-picture and recent history of emotions.

1.2 Objectives

1. An attractive and easy to use application for entertainment purpose.

SER features a sleek and intuitive user interface, designed for an engaging and enjoyable experience. Its vibrant colors and user-friendly layout make it visually appealing and easy to navigate, ensuring a delightful user interaction.

2. Pose as a friend, converse with user with appropriate response

SER chatbot employs natural language processing to engage users in friendly and lifelike conversations, making them feel like they are chatting with a real friend.

3. Display detected emotions on screen.

Users can easily see how their emotions are interpreted, with the help of creative pictures such as mic on, mic off and hourglass creating an interactive and entertaining aspect of the application.

4. Detect emotions accurately.

SER utilises advanced machine learning algorithms to accurately identify and classify emotions in speech, providing a reliable assessment of emotional state.
It takes into account various vocal cues, such as chroma, mel, MFCC and speech patterns, to ensure precise emotion detection, enhancing the overall entertainment experience.

5. Proper exception handling.

Our application is equipped with robust error handling mechanisms to gracefully handle unexpected situations, ensuring a smooth and uninterrupted user experience.

1.3 Purpose and Scope

1.3.1 Purpose

1. The primary purpose of this application is to create a genuinely interactive and emotionally intelligent chatbot that enhances user engagement by recognizing and responding to their emotions effectively.
2. By recognizing and displaying emotions from user queries, the application aims to provide a more personalized and empathetic conversational experience.
3. It serves as a tool for fostering emotional well-being, as it encourages users to express their feelings and thoughts, potentially leading to better self-awareness and emotional regulation.
4. It can be a fun and entertaining platform for users to engage with, as the chatbot's ability to mirror emotions adds a new dimension to entertainment and interactive storytelling.
5. In educational settings, the application can be used as a tool for teaching emotional intelligence and effective communication by showcasing the importance of understanding and responding to emotions in conversations.
6. The application can also be used for research and data collection, enabling the analysis of emotional trends and patterns in user interactions.
7. By promoting emotionally responsive AI, the application contributes to the development of AI systems that better understand and cater to emotional needs.
8. It offers a safe space for users to share their feelings and receive non-judgmental responses, which can be particularly beneficial in situations where human interaction may be limited.
9. Overall, the primary purpose of this application is to harness the power of AI to create a more emotionally aware, empathetic, and engaging chatbot, facilitating deeper connections and personal growth while also providing entertainment and support to users.

1.3.2 Scope

1. The scope of this project encompasses the development of a sophisticated chatbot with emotion recognition capabilities.
2. It will include the integration of natural language processing and machine learning techniques to enable the chatbot to understand and respond to a wide range of user queries and emotions.
3. The chatbot's emotional recognition scope will cover a diverse set of emotional states such as happy, angry, neutral, sad.
4. The application will employ real-time emotion detection, ensuring that user emotions are continuously monitored and displayed on the screen during conversations.
5. It will have a user-friendly interface that allows users to easily interact with the chatbot and observe their emotions reflected in real-time.
6. The chatbot will be designed to provide appropriate responses that match the detected emotions, creating a dynamic and emotionally responsive conversational experience.
7. The project's overall scope aims to create an emotionally intelligent chatbot that can be applied in various contexts, such as mental health support, entertainment, education, and research, with the potential for expansion into additional domains in the future.

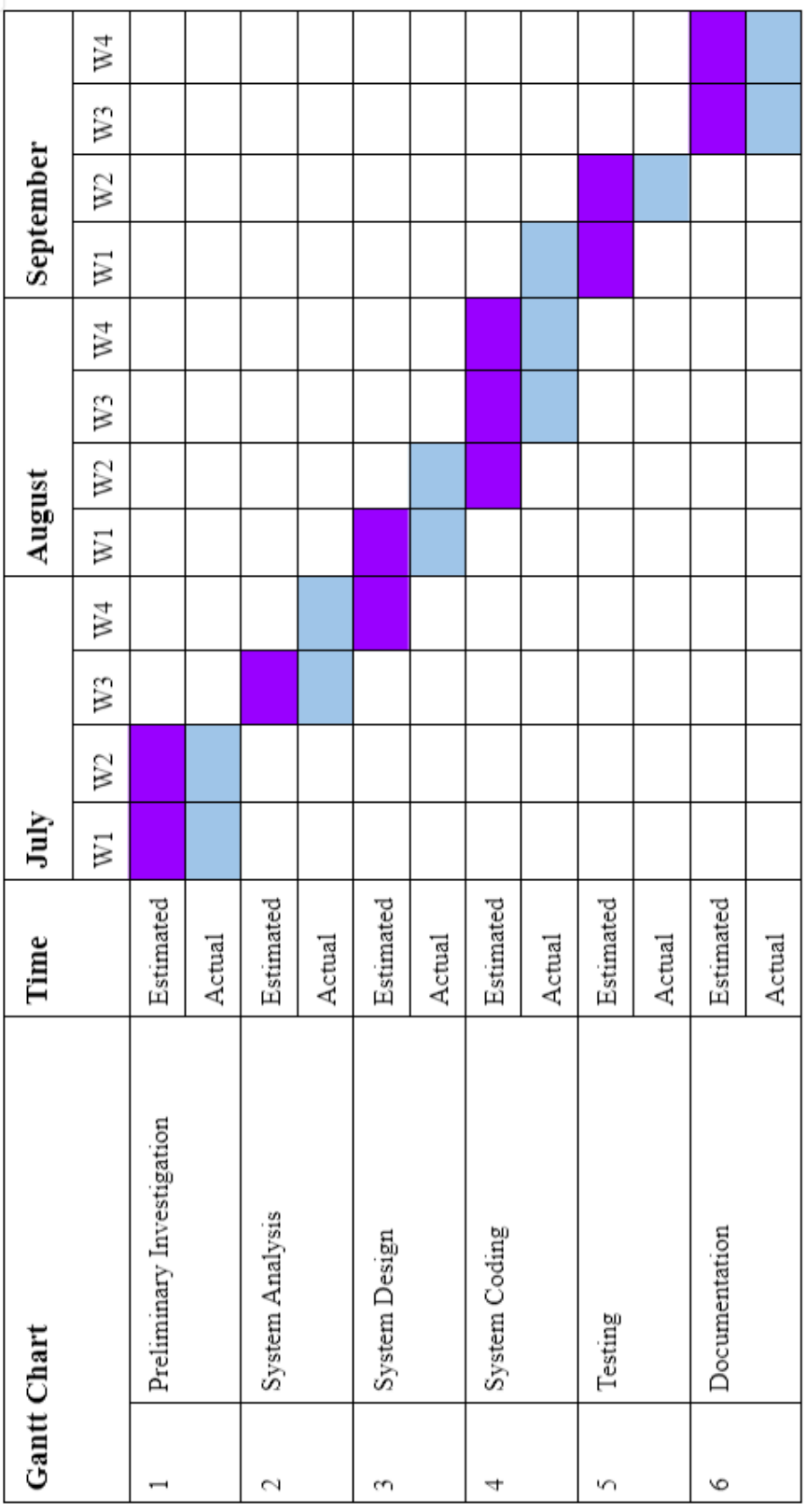
Future Scope

1. Chat bot witll use appropriate emojis and punctuation in queries and responses
2. Continous learning algorithm that stores all entries of queries and machine evaluated emotion and prompt user to validate whether the detected emotion was correct or not. This data will then be used to enhance the current model.

1.4 Phase Title

Phase Title	Expected Date of Completion	Actual Time of Completion with Guide's Signature	Remarks
I. Preliminary Investigation			
(i) Project Overview	20/08/2023		
(ii) Feasibility Study			
(iii) Phase Title			
(iv) Gantt Chart			
II. System Analysis			
(i) Existing System	09/09/2023		
(ii) Proposed System			
(iii) System Requirements			
III. System Design			
(i) Flow Chart	15/09/2023		
(ii) Use Case Diagram			
(iii) Sequence Diagram			
(iv) ER Diagram			
IV. System Coding			
(i) Model Building	25/09/2023		
(ii) System Coding			
(iii) Testing Approaches			
V. Future Enhancements			
VI. References	25/10/2023		

1.5 Gantt Chart



Gantt Chart		Time	July				August				September			
			W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4
1	Preliminary Investigation	Estimated												
		Actual												
2	System Analysis	Estimated												
		Actual												
3	System Design	Estimated												
		Actual												
4	System Coding	Estimated												
		Actual												
5	Testing	Estimated												
		Actual												
6	Documentation	Estimated												
		Actual												

CHAPTER 2

SYSTEM ANALYSIS

2.1 Existing System

The existing system of chatbots that rely solely on written prompts without incorporating emotion recognition techniques typically offers a more limited and less empathetic conversational experience. These traditional chatbots primarily follow scripted responses based on keywords or predefined patterns, which can lead to less contextually relevant and emotionally disconnected interactions.

Firstly, these chatbots often struggle to comprehend the emotional nuances present in user queries. They are unable to detect changes in tone, mood, or emotional context, which can result in responses that may seem out of touch with the user's actual state of mind. As a result, users may find it challenging to establish a genuine emotional connection with such chatbots, limiting their potential for providing emotional support or understanding.

Secondly, without emotion recognition capabilities, these chatbots miss out on an essential aspect of human communication. Human conversations are not solely based on text; they also involve vocal cues, body language, and emotional inflections that play a crucial role in conveying the full spectrum of human emotions. By neglecting emotion recognition, these chatbots fail to engage with users on a deeper level, potentially hindering their utility in areas like mental health support, where emotional understanding is crucial.

In conclusion, the existing system of chatbots that lack emotion recognition techniques may serve practical purposes for basic queries and tasks but falls short in delivering emotionally intelligent and empathetic interactions.

2.1 Proposed System

The proposed system of a chatbot that utilises machine learning to detect emotions in a user's query (speech) and displays them on the screen while providing appropriate responses represents an exciting and promising advancement in AI-driven communication. In this innovative approach, emotion recognition and natural language processing (NLP) operate in parallel, allowing for a more comprehensive understanding of emotional state and textual input.

Firstly, the integration of emotion detection in a chatbot's functionality is a significant step toward achieving more empathetic and context-aware interactions. By using machine learning models to analyze vocal cues, such as chroma, Mel, MFCC and speech patterns, the chatbot can accurately identify and display the user's emotions on the screen. This visual representation of emotions enhances user awareness of their own feelings during the conversation, fostering emotional self-reflection.

Secondly, by running emotion recognition and NLP in parallel, the chatbot lays the groundwork for the development of a more emotionally responsive conversational agent. While not yet integrated, the future potential is evident in that these parallel systems can collaborate to provide tailored and empathetic responses. This innovative design allows for flexibility, as the chatbot can continue to respond contextually based on the text input while concurrently acknowledging the user's emotions through on-screen displays.

In conclusion, the proposed chatbot system, with parallel operation of emotion recognition and NLP, shows great promise in enhancing the user experience by recognizing and reflecting emotions in real time.

2.3 Requirement Analysis

2.3.1 Hardware Requirements

Component	Recommended Specifications
Processor	Modern multi-core processor (2.0 GHz or higher)
RAM	At least 8 GB

2.3.1 Software Requirements

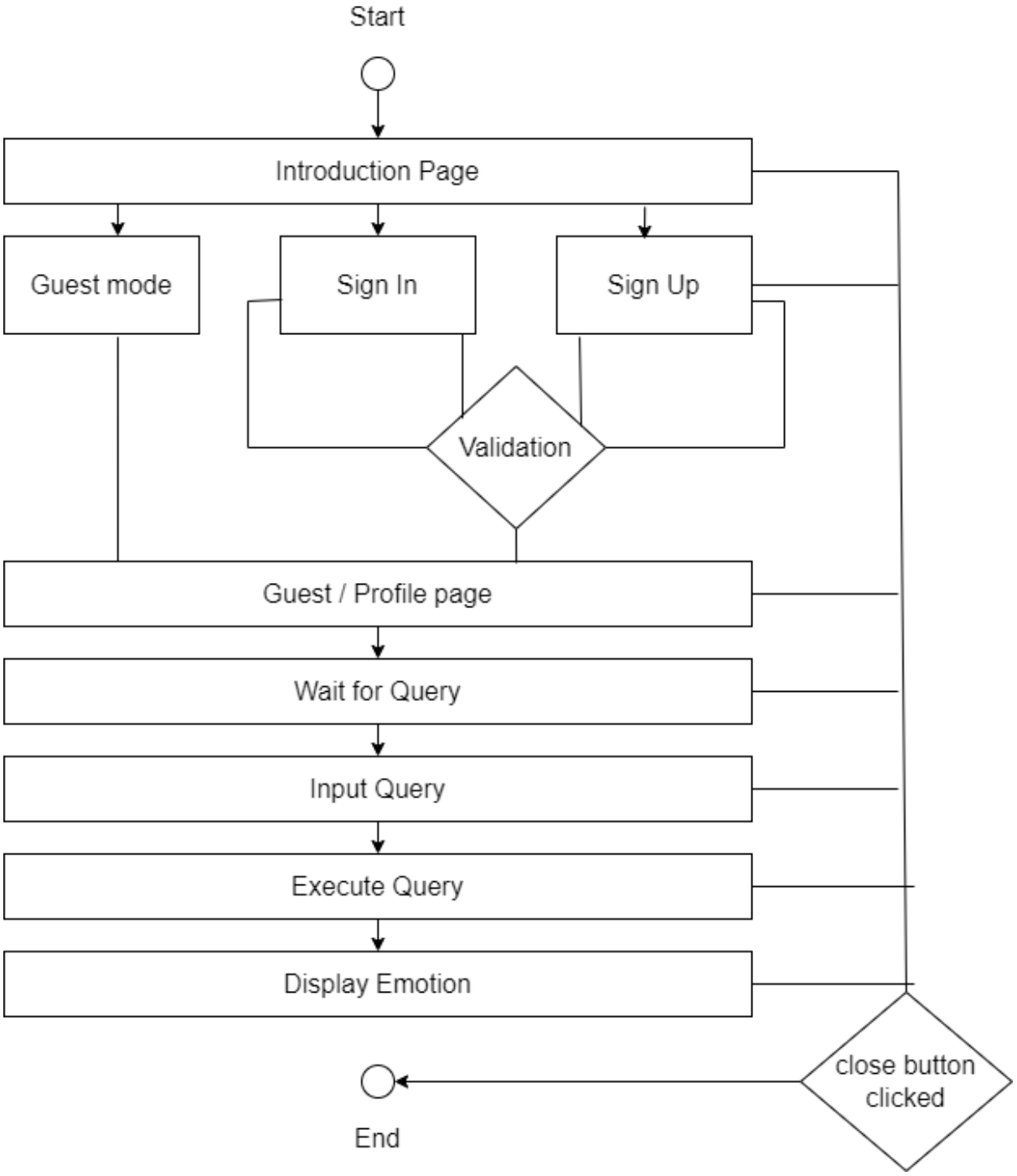
Software Component	Required Version
Python	Python 3.7 or higher
Python Libraries	Librosa, Pickle, Tkinter, CustomTkinter
Machine Learning Libraries	scikit-learn, numpy, pandas
Machine Learning Algorithm	Multi-layer Perceptron classifier
Openai API	API key from Openai.com

\

CHAPTER 3

SYSTEM DESIGN

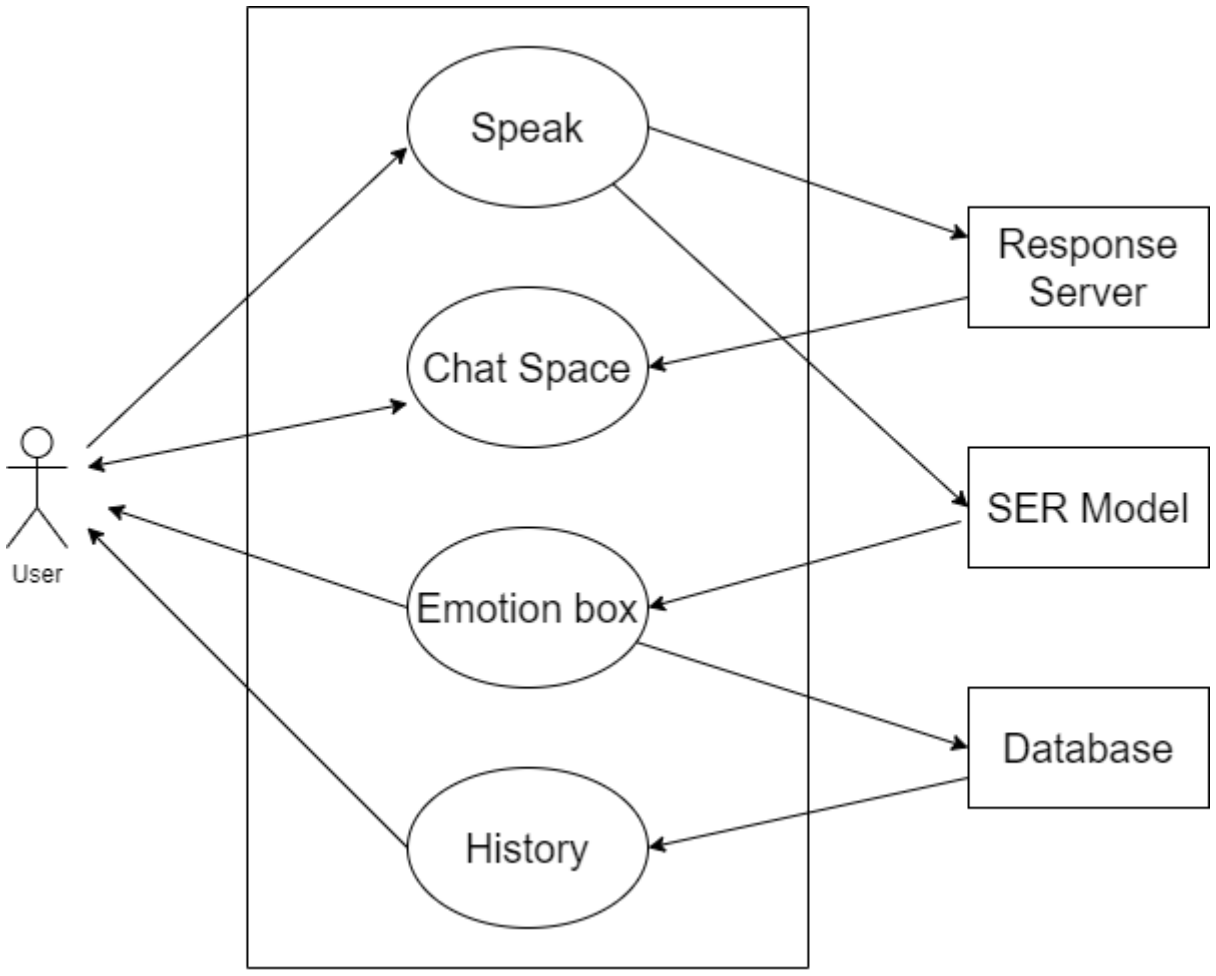
3.1 System Flow Chart



3.1 Flow Chart

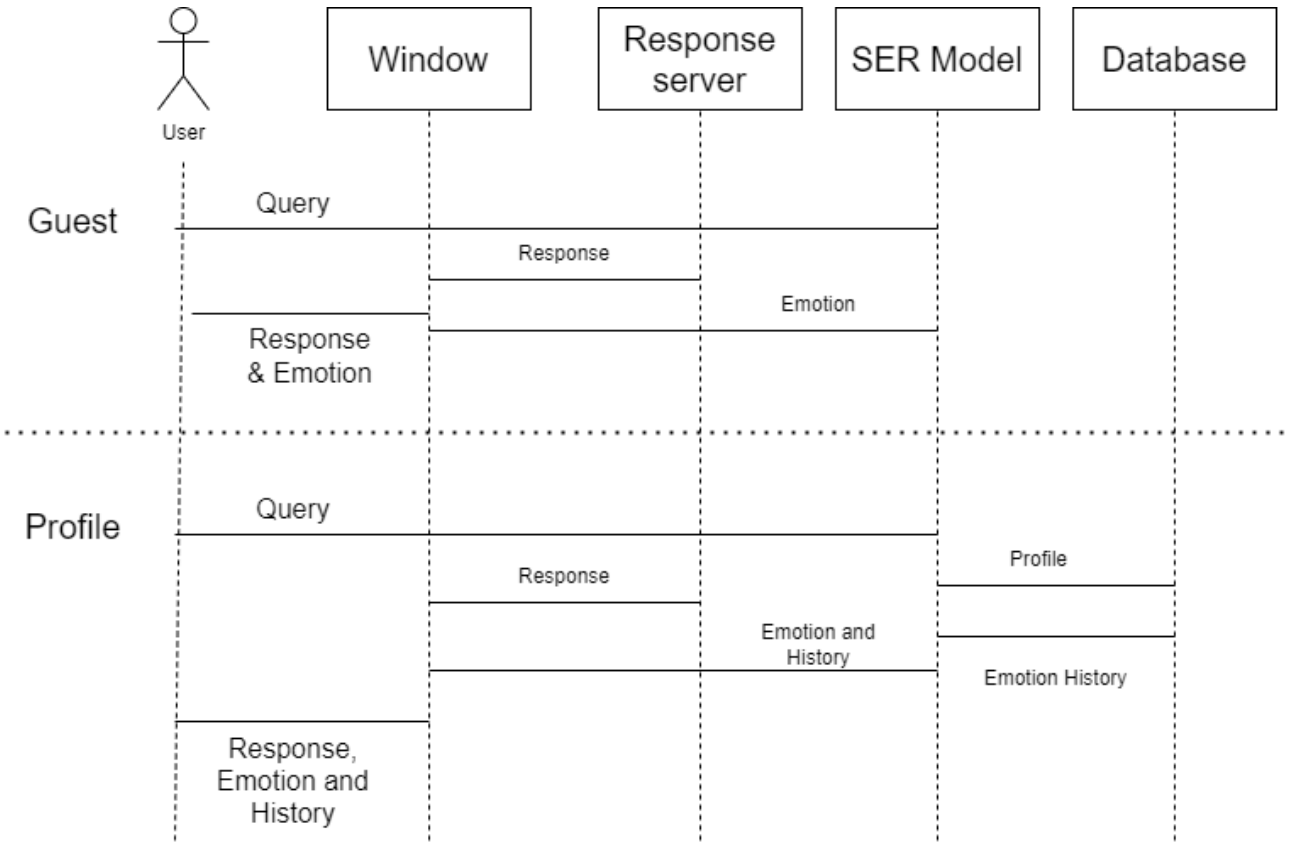
3.2 UML Diagrams

3.2.1 Use Case Diagram



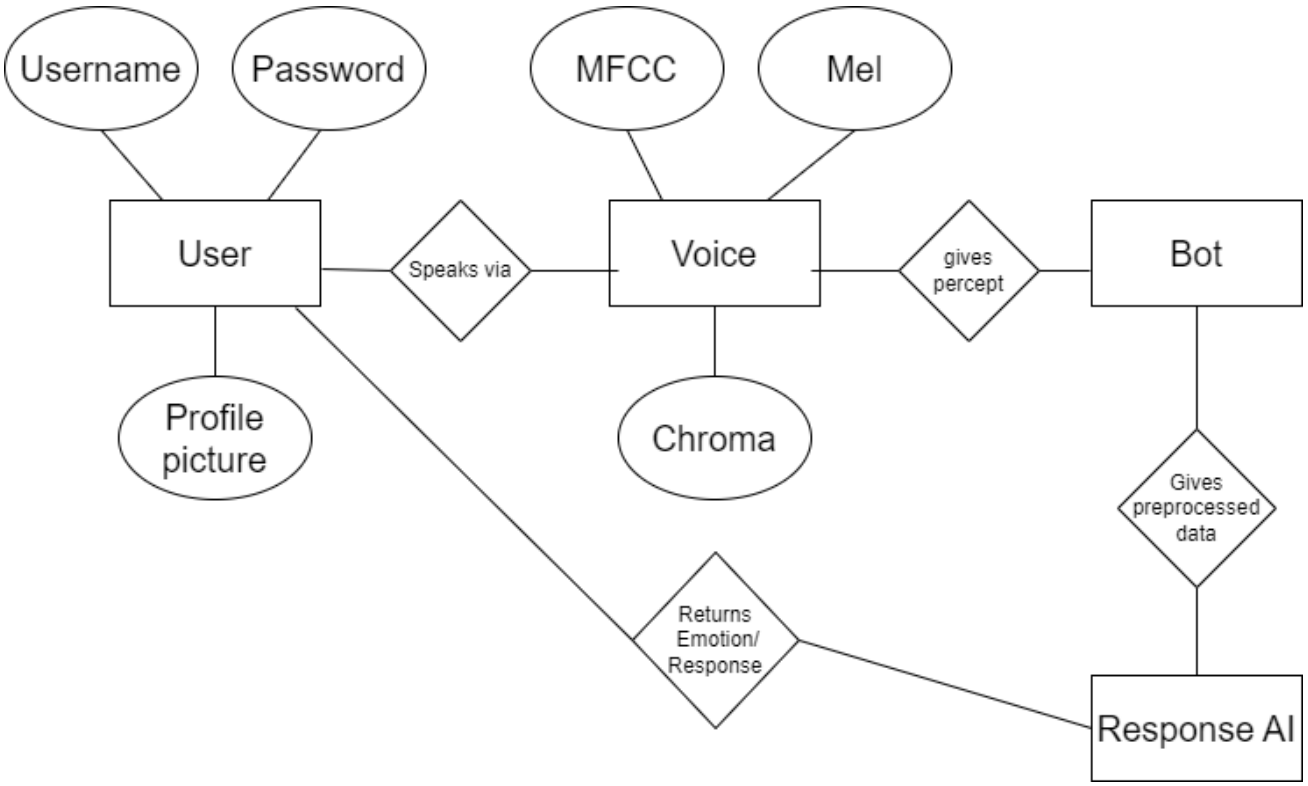
3.2.1 Use Case Diagram

3.2.2 Sequence Diagram



3.2.1 Sequence Diagram

3.2.3 Entity Relationship Diagram



3.2.3 Entity Relationship Diagram

CHAPTER 4

IMPLEMENTATION AND TESTING

4.1.1 SER Model
Jupyter Notebook File(ipnb file)

In [1]:
```
import librosa
import soundfile
import os, glob, pickle
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.neural_network import MLPClassifier
from sklearn.metrics import accuracy_score
```

In [2]:
```
#DataFlair - Extract features (mfcc, chroma, mel) from a sound file
def extract_feature(file_name, mfcc, chroma, mel):
    with soundfile.SoundFile(file_name) as sound_file:
        X = sound_file.read(dtype="float32")
        sample_rate=sound_file.samplerate
        if chroma:
            stft=np.abs(librosa.stft(X))
        result=np.array([])
        if mfcc:
            mfccs=np.mean(librosa.feature.mfcc(y=X, sr=sample_rate, n_mfcc=40).T
            result=np.hstack((result, mfccs))
        if chroma:
            chroma=np.mean(librosa.feature.chroma_stft(S=stft, sr=sample_rate).T
            result=np.hstack((result, chroma))
        if mel:
            mel=np.mean(librosa.feature.melspectrogram(y=X, sr=sample_rate).T,ax
            result=np.hstack((result, mel))
    return result
```

In [3]:
```
#DataFlair - Emotions in the RAVDESS dataset
emotions={
  '01':'neutral',
  '02':'calm',
  '03':'happy',
  '04':'sad',
  '05':'angry',
  '06':'fearful',
  '07':'disgust',
  '08':'surprised'
}
#DataFlair - Emotions to observe
#observed_emotions=['calm', 'happy', 'fearful', 'disgust']
observed_emotions=['neutral', 'happy', 'sad', 'angry']
```

In [4]:
```
#DataFlair - Load the data and extract features for each sound file
def load_data(test_size=0.2):
    x,y=[],[]
    for file in glob.glob("C:\\Users\\Admin\\Desktop\\SER\\Ravdass_lfr\\Actor_*\
        file_name=os.path.basename(file)
        emotion=emotions[file_name.split("-")[2]]
        if emotion not in observed_emotions:
            continue
        feature=extract_feature(file, mfcc=True, chroma=True, mel=True)
        x.append(feature)
        y.append(emotion)
    return train_test_split(np.array(x), y, test_size=test_size, random_state=9)
```

In [5]:
```
#DataFlair - Split the dataset
x_train,x_test,y_train,y_test=load_data(test_size=0.25)
```

In [6]:
```
#DataFlair - Get the shape of the training and testing datasets
print((x_train.shape[0], x_test.shape[0]))
```

```
(504, 168)
```

In [7]:
```
#DataFlair - Get the number of features extracted
print(f'Features extracted: {x_train.shape[1]}')
```

```
Features extracted: 180
```

In [8]:
```
#DataFlair - Initialize the Multi Layer Perceptron Classifier
model=MLPClassifier(alpha=0.01, batch_size=256, epsilon=1e-08, hidden_layer_size
```

In [9]:
```
#DataFlair - Train the model
model.fit(x_train,y_train)
```

Out[9]:

MLPClassifier

```
MLPClassifier(alpha=0.01, batch_size=256, hidden_layer_sizes=(300,),
              learning_rate='adaptive', max_iter=500)
```

In [10]:
```
#DataFlair - Predict for the test set
y_pred=model.predict(x_test)
```

In [11]:
```
#DataFlair - Calculate the accuracy of our model
accuracy=accuracy_score(y_true=y_test, y_pred=y_pred)
#DataFlair - Print the accuracy
print("Accuracy: {:.2f}%".format(accuracy*100))
```

```
Accuracy: 70.24%
```

In [12]:
```
pickle.dump(model, open('model1.pkl','wb'))
```

4.1.2 Code

Imports:

```
import time
import customtkinter as ct
import PIL.Image
import datetime
import openai
import tkinter.scrolledtext as ScrolledText
from tkinter import *
import pyttsx3
import speech_recognition as sr
import os
from requests import get
import wikipedia
import webbrowser
import threading
import librosa
import soundfile
import numpy as np
import pickle
import tempfile
from time import sleep
import mysql.connector
import sklearn.model_selection
import sklearn.neural_network
import sklearn.metrics
```

Setting up OpenAI response server:

```
openai.api_key =
'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'
messages = [
    {"role": "system", "content": "You are a kind helpful assistant."},
]
```

Setting up color schemes:

```
default_light = "#a3cfcf"
```

```
default_dark = "#3857e0"
ct.set_appearance_mode("light")
ct.set_default_color_theme("blue")

red_light = "#fca5a2"
red_dark = "#fc0e05"
blue_light = "#a3cfcf"
blue_dark = "#3857e0"
yellow_light = "#fafca4"
yellow_dark = "#d3d61e"
green_light = "#77fa73"
green_dark = "#0b7a01"
pink_light = "#fccff8"
pink_dark = "#fc03c1"
purple_light = "#dcadff"
purple_dark = "#a103fc"
```

Setting Voice Recognition:

```
engine = pyttsx3.init('sapi5')
voices = engine.getProperty('voices')
engine.setProperty('voice', voices[0].id)
engine.runAndWait()
r = sr.Recognizer()
r.pause_threshold = 1
```

Emotion Recognition and Prediction:

```
def extract_feature(file_name, mfcc, chroma, mel):
   with soundfile.SoundFile(file_name) as sound_file:
      X = sound_file.read(dtype="float32")
      sample_rate=sound_file.samplerate
      if chroma:
         stft=np.abs(librosa.stft(X))
      result=np.array([])
      if mfcc:
         mfccs=np.mean(librosa.feature.mfcc(y=X, sr=sample_rate,
n_mfcc=40).T, axis=0)
         result=np.hstack((result, mfccs))
      if chroma:
```

```
            chroma=np.mean(librosa.feature.chroma_stft(S=stft,
sr=sample_rate).T,axis=0)
            result=np.hstack((result, chroma))
        if mel:
            mel=np.mean(librosa.feature.melspectrogram(y=X,
sr=sample_rate).T,axis=0)
            result=np.hstack((result, mel))
    return result

# Load the trained model for speech emotion recognition
def load_emotion_model():
    script_dir = os.path.dirname(os.path.abspath(__file__))
    model_path = os.path.join(script_dir, 'model1.pkl')
    #model_path = "model1.pkl"  # Replace with the actual path
    with open(model_path, 'rb') as model_file:
        model = pickle.load(model_file)
    return model

emotion_model = load_emotion_model()

# Function to predict emotion from user's voice input
def predict_emotion(audio_data):
    feature = extract_feature(audio_data, mfcc=True, chroma=True, mel=True)
    emotion = emotion_model.predict([feature])[0]
    return emotion

def set_text(e, text):
    e.delete(0,END)
    e.insert(0,text)
    return

def listen(mic, listening, hourglass, photo, emotion_label, entry, ChatLog,
usrnme):
    global query
    with sr.Microphone() as source:
        photo.configure(image=listening)
        audio = r.listen(source, phrase_time_limit=5)
    try:
        photo.configure(image=hourglass)
        query = r.recognize_google(audio, language="en-in")
        print("log: recognized")
```

```
        #query_label.configure(text=f"User said: {query}")
        set_text(entry, query)
        enter_thread(entry, ChatLog, photo.cget("bg_color"))
        print("log: enter function done")

        # Convert audio frame_data to NumPy array of int16
        audio_data = np.frombuffer(audio.frame_data, dtype=np.int16)

        # Save the audio data as a temporary WAV file
        temp_wav_path = tempfile.NamedTemporaryFile(suffix=".wav",
delete=False).name
        soundfile.write(temp_wav_path, audio_data, audio.sample_rate)
        predicted_emotion = predict_emotion(temp_wav_path)
        sleep(4)
        # Creating connection object
        mydb = mysql.connector.connect(
            host="localhost",
            user="root",
            password="test1",
            autocommit=True
        )
        cursor = mydb.cursor()
        cursor.execute("use projectdb;")
        if predicted_emotion:
            emotion_list.append(predicted_emotion)
            print(emotion_list)
            # emo_string = ",".join(emotion_list)
            emo_string = ", " + emotion_list[-1]
            #emo_query = "select emotions from logindata ORDER BY id DESC
LIMIT 1;"
            emo_query = f"select emotions from logindata where
username='{usrnme}';"
            cursor.execute(emo_query)
            print("1")
            value = cursor.fetchone()
            print("2")
            full_string = value[0] + emo_string
            print(full_string)
            if len(full_string)>230:
                full_string = full_string[10:]
            update_query = f"update logindata set emotions='{full_string}' where
username='{usrnme}';"
```

```
            cursor.execute(update_query)
            print("3")
            emotion_label.configure(text="Emotion: "+predicted_emotion)
            print(f"Emotion: {predicted_emotion}")


        else:
            set_text(emotion_label, "Not Recognized")
            print("Not Recognized")

        # Clean up temporary audio file
        os.remove(temp_wav_path)
        photo.configure(image=mic)

        return query
    except Exception as e:
        print("Exception.")
        photo.configure(image=mic)


# seperate thread for listening
def listen_thread(mic, listening, hourglass, photo, emotion_label, entry,
ChatLog, usrnme):
    thread1 = threading.Thread(target=lambda:listen(mic, listening, hourglass,
photo, emotion_label, entry, ChatLog, usrnme))
    thread1.start()

# bot speak
def speak(audio):
  print(audio)
  engine.say(audio)
  engine.runAndWait()

def exec_query(query):
    if "open notepad" in query:
        npath = "C:\\Windows\\system32\\notepad"
        os.startfile(npath)
        result = "anything else?"
        return result
    elif "open command prompt" in query:
        os.system("start cmd")
        result = "anything else?"
```

```
    return result
elif "ip address" in query:
    ip = get("http://api.ipify.org").text
    result = f"your IP address is {ip}\nanything else?"
    return result
elif "wikipedia" in query:
    query = query.replace("wikipedia", "")
    result = wikipedia.summary(query, sentences=2)
    result = f"according to wikipedia...{result}\nanything else?"
    return result
elif "open youtube" in query:
    webbrowser.open("www.youtube.com")
    result = "anything else?"
    return result
elif "no thanks" in query:
    result = "thanks for using me, have a good day"
    return result
else:
    result = response(query)
    return result
```

4.2 Testing Approach

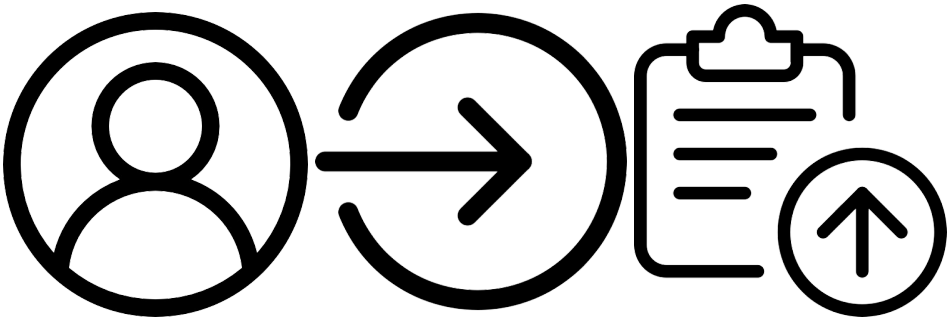
- Unit test cases were tested manually.
- After all unit tests passed the test, overall exploratory tests were done by third-person.

Test Cases	Expected Result	Actual Result	Status
Navigations	All navigations should redirect to appropriate page.	All navigation redirect to appropriate page.	Pass
Sign Up Validation	Empty username and password should not be allowed. User should be informed ‘username can’t be empty’ when trying to proceed with empty entries.	Empty username and password is not allowed. User is informed ‘username can’t be empty’ when trying to proceed with empty entries.	Pass
Sign In Validation	Incorrect username and passwords should not be allowed. User should be informed what is incorrect (username/password).	Incorrect username and passwords are not allowed. User is informed what is incorrect (username/password).	Pass
Enter	When chatting via keyboard, pressing ‘enter’ key message should be sent.	When chatting via keyboard, on pressing ‘enter; key message is sent.	Pass
Speak	On pressing the Speak button, mic should be activated.	On pressing the Speak button, mic is activated.	Pass
Colour	On pressing a colour button, the respective colour scheme should be applied.	On pressing a colour button, the respective colour scheme is applied.	Pass

CHAPTER 5

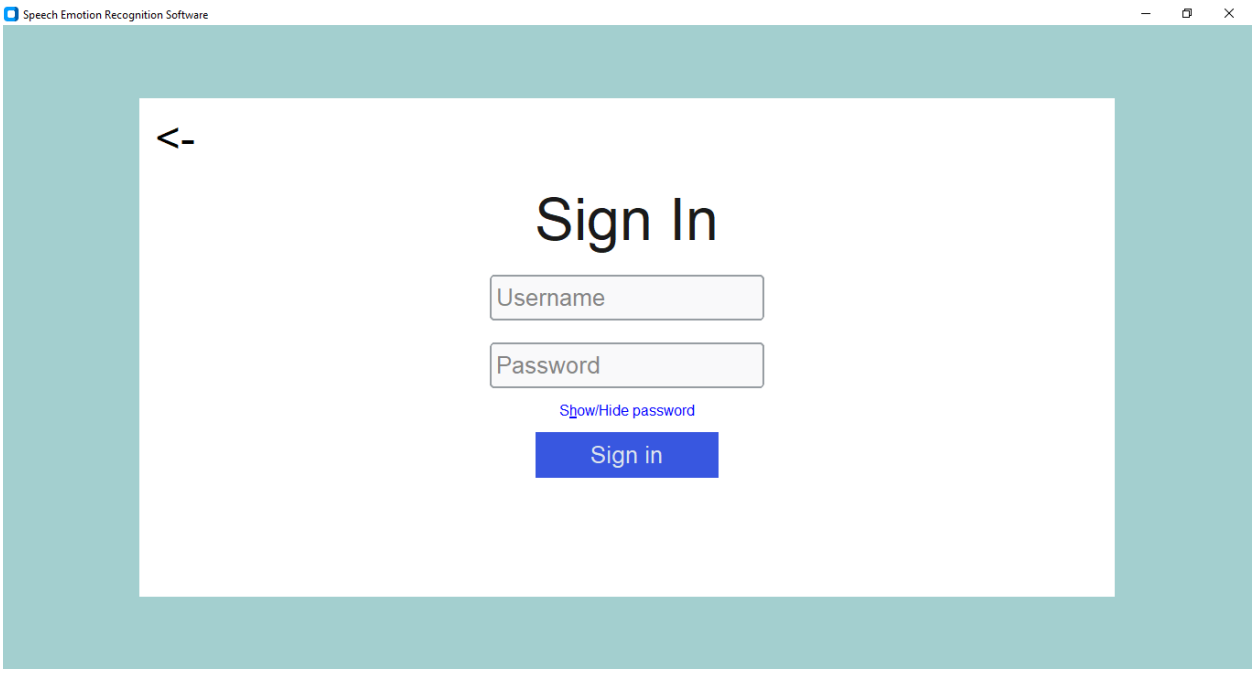
RESULTS

Images used:

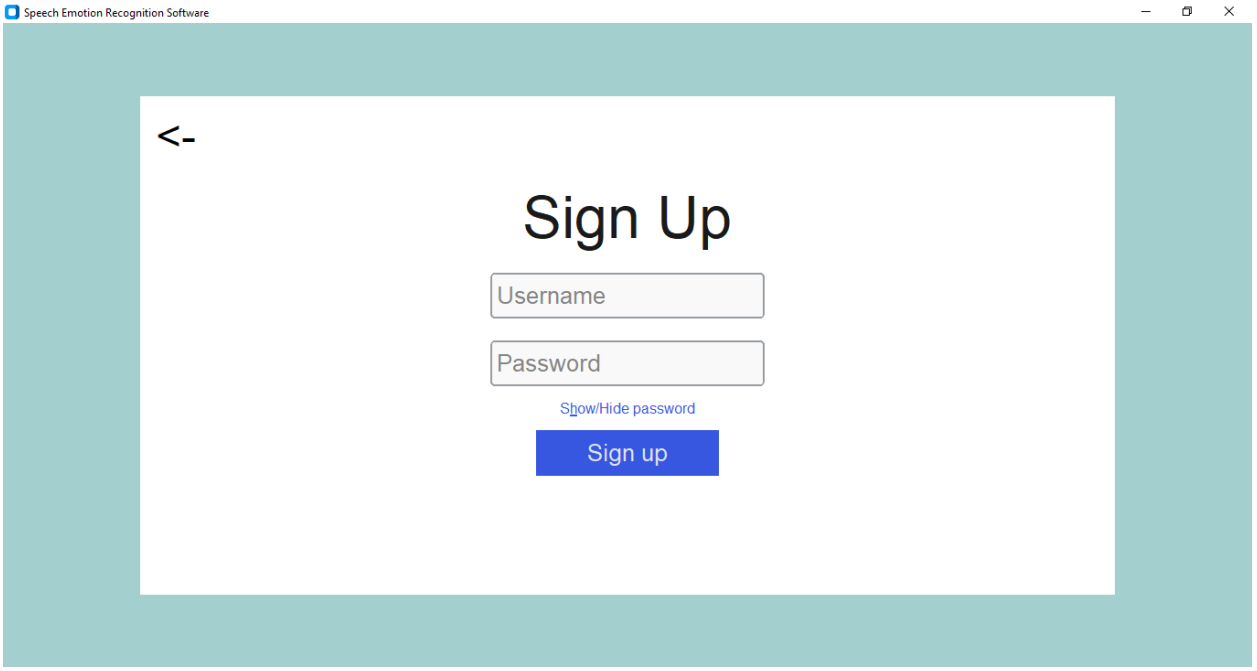


5.1 Splash screen

5.2 Guest Mode

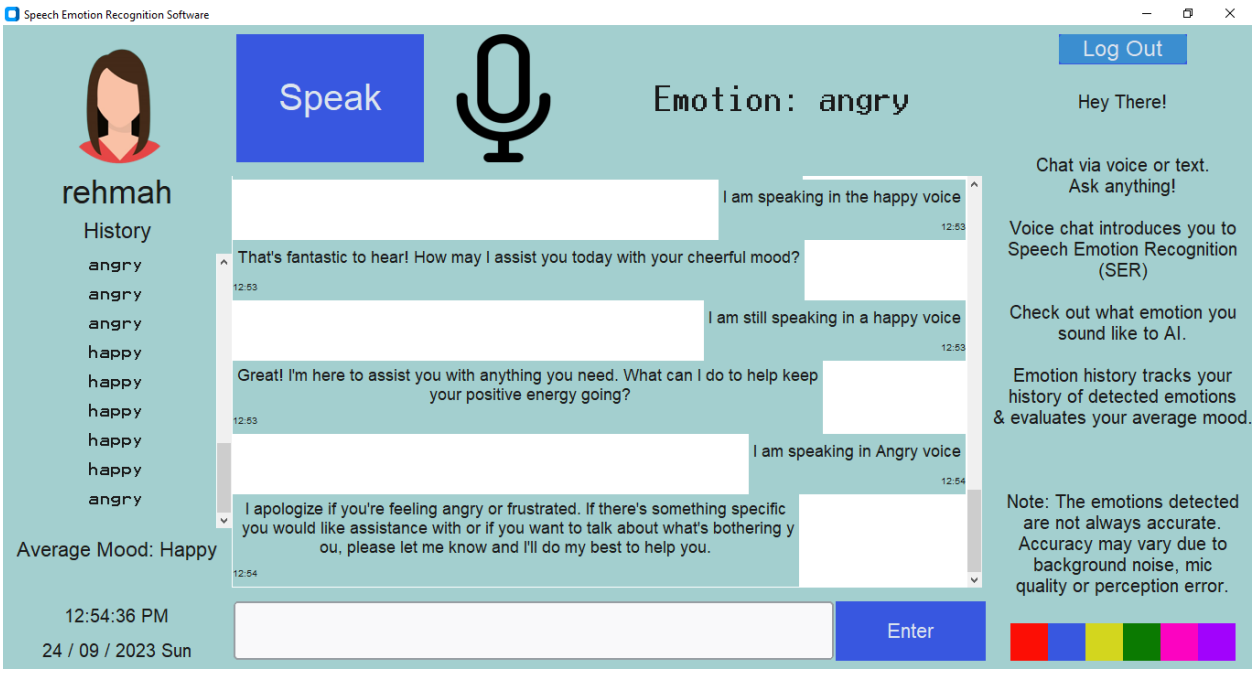


5.3.1 Sign In



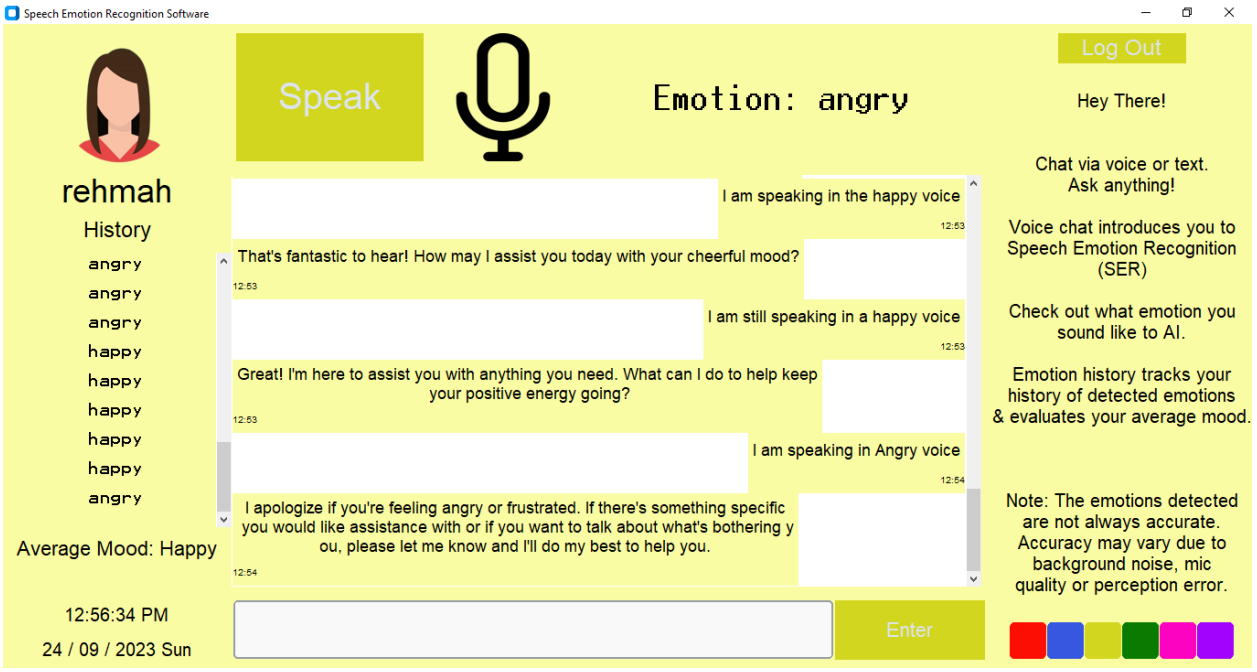
5.3.2 Sign Up

Profile Mode and Themes

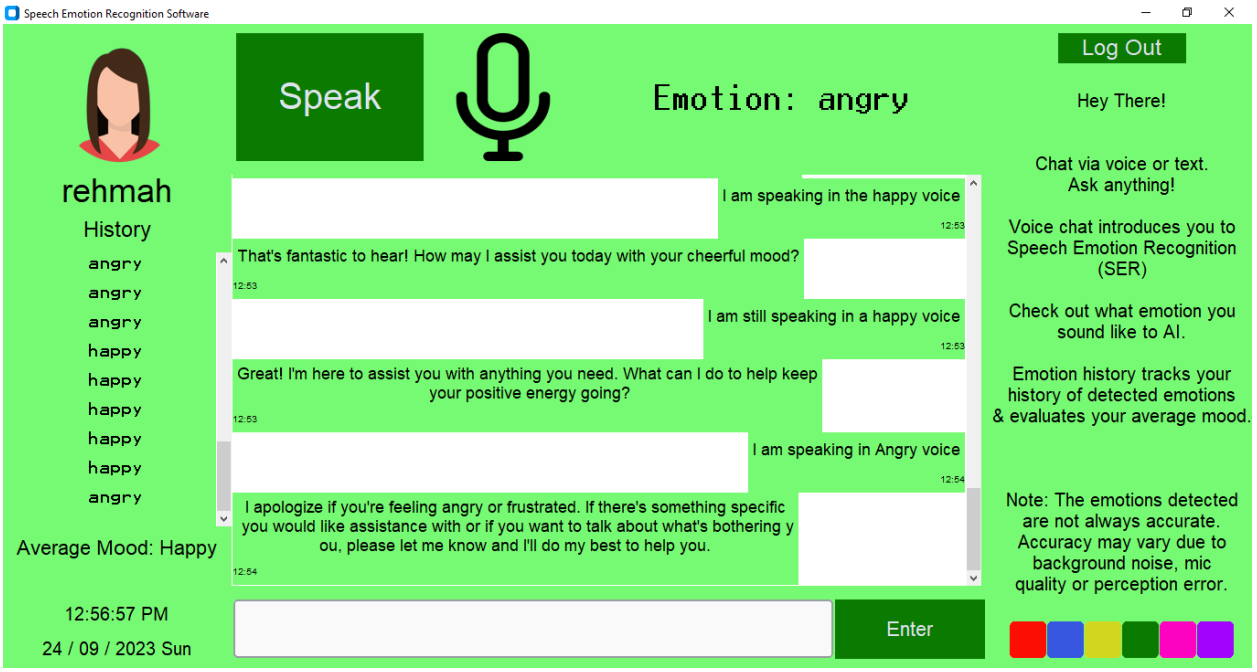


5.4.1 Sky theme

5.4.2 Cherry theme

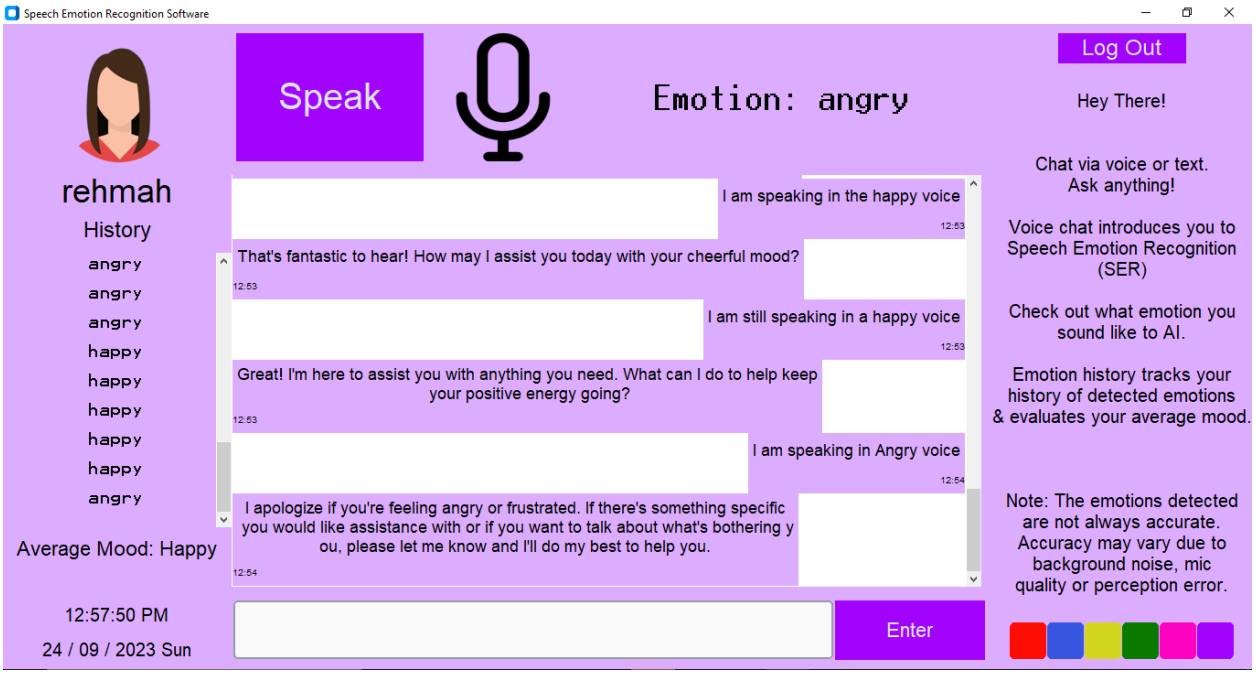


5.4.3 Lime theme



5.4.4 Grass theme

5.4.5 Rose theme



5.4.6 Amethyst theme

CHAPTER 6

CONCLUSION AND FUTURE WORK

Conclusion

In conclusion, this documentation provides a comprehensive overview of the proposed project, a Speech Emotion Recognition Software (SERS), designed to enhance human-computer interactions by recognizing and displaying emotions in user queries while offering appropriate responses. The integration of emotion detection and NLP in parallel represents an innovative approach that has the potential to provide users with a more meaningful and emotionally aware conversational experience.

The project's objectives encompass creating an attractive and user-friendly application for entertainment purposes, enabling the chatbot to pose as a friendly conversational partner, displaying detected emotions on-screen, and ensuring accurate emotion detection. Additionally, the system incorporates proper exception handling to ensure a smooth and uninterrupted user experience. Its purposes span from entertainment and emotional well-being to education, research, and emotional intelligence development.

The scope of the project covers a wide range of features, including machine learning-based emotion recognition, a user-friendly interface, and emotionally responsive chatbot interactions, all with potential applications in various domains. The project also outlines future scopes for emoji integration and continuous learning to enhance the system further. Overall, the Speech Emotion Recognition Software has the potential to offer users an emotionally intelligent and engaging conversational experience, fostering deeper connections, personal growth, and entertainment while contributing to the advancement of AI systems that understand and cater to emotional needs.

Future Work

While the current Speech Emotion Recognition system parallel with NLP response feature have proven to be highly effective, there are several areas for future enhancement that could further improve its capabilities.

1. Continuous Learning

This system can be further enhanced by adding feedback from user on each query whether the detected emotion was correct or not, the software should then store audio features and their evaluated emotion along with user validated emotion then using this data for further enhancement of training model.

2. Higher Accuracy

The SER model can be further enhanced by adding more relevant features and techniques to it. Higher accuracy will highly elevate the systems functionality and provide user satisfaction.

3. Integration of NLP with SER instead of working parallely

Over all the current system only provides general user interactive environment where SER and NLP works separately but parallel. A impactful step will be to integrate SER weights with NLP, the responses should take into account the detected emotion for response formulation. This could be done by classifying NLP patterns into classes based on each emotion, thus when emotion is detected a response is fetched from its respective class. This is only one way to do it, further more detailed and much more interactive NLP model can be developed customised by emotion classes.

CHAPTER 7

REFERENCES

References

[1] Model Guidance - Dataflair : (https://data-flair.training/blogs/python-mini-project-speech-emotion-recognition/)

[2] Dataset - RAVDASS (Ryerson Audio-Visual Database of Emotional Speech &Song):(https://www.kaggle.com/datasets/uwrfkaggler/ravdess-emotional-speech-audio)

[3] API- Openai API : (https://openai.com/blog/openai-api)

[4] Speech Emotion Recognition: A survey by Kunal Bhapkar1, Krati Patni2, Praddyumn Wadekar3, Shweta Pal4, Dr. Rubeena A. Khan5, Mahesh Shinde6 at International Research Journal of Engineering and Technology (IRJET) on/in Volume: 08 Issue: 03 | Mar 2021

[5] Article on Speech Emotion Recognition using Python | Sound Classification | Deep Learning at hackersrealm.net, uploaded in June 2022.

[6] Research Paper on “Speech based Emotion Recognition using Machine Learning” by Sukanya anil Kulkarni and her fellow mates at 3rd International Conference on Computing Methodologies and Communication (ICCMC) in 2019

[7] Research Paper on “Machine Learning Based Speech Emotions Recognition System” by Dr. Yogesh Kumar, Dr. Manish Mahajan in INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 8, ISSUE 07, JULY 2019

Plagiarism Report

Plagiarism Scan Report

Report Generated on: Nov 03,2023

Plagiarised

Unique

Total Words:	996
Total Characters:	6667
Plagiarized Sentences:	6.72
Unique Sentences:	41.28 (86%)

Content Checked for Plagiarism

1. Introduction

Emotions help us to communicate with others, such as when we feel sad and need some help. They also can help us to act quickly in important situations. They can help you survive, grow, and connect with others. And they can guide your decisions, behaviors, and motivations. As babies, emotions are how you learn to communicate, even before you can talk.

In a world of cutting edge technology and its constant interaction with human beings, a seamless communication is a must to get desired response. While Natural Language processing (NLP) has made AI-Human interaction smooth, it leaves room for perception of unworded emotion.

Advances in machine learning and artificial intelligence offer promising solutions to bridge this gap by providing predictive models, guidance on preventive measures, and assistance in connecting with healthcare providers efficiently.

This project harnesses the power of machine learning, to predict emotions based on user's queries. Additionally, it offers a friendly chat-bot that is well equipped with NLP to provide perfect responses and solutions to any prompt.

Speech Emotion Recognition Software (SERS) is a smart AI tool that can converse via speech or text, respond well to any prompts and questions, perceive voice features and evaluate emotion behind one's voice.

1.1 Background and Project Overview

The project is purely python coded. The emotion detection system is based on the ML model shown in the following section. The model is trained on Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS). The dataset consists of 24 actors(12 Male, 12 Female) recordings of 8 emotions.

The model uses preprocessed dataset that is trained on 4 emotions that are Happy, Neutral, Sad and Angry. The model checks for features such as Chroma, Mel, MFCC (Mel-frequency cepstral coefficients) to train with the help of MLPClassifier (Multi-layer Perceptron classifier) which is a model that relies on an underlying Neural Network to perform the task of classification.

Prediction features are extracted with the help of python's Librosa library. Librosa is valuable Python music and sound investigation library that helps programming designers to fabricate applications for working with sound and music.

For Chatbot and NLP, the project uses Openai's API. The user prompts are sent to Openai servers and a

fitting response is sent back to the software with the help of API key.

For profiles, login data and emotion history the software uses SQL database. The database stores usernames, passwords, profile-picture and recent history of emotions.

1.2 Objectives

An attractive and easy to use application for entertainment purpose.
SER features a sleek and intuitive user interface, designed for an engaging and enjoyable experience. Its vibrant colors and user-friendly layout make it visually appealing and easy to navigate, ensuring a delightful user interaction.

Pose as a friend, converse with user with appropriate response
SER chatbot employs natural language processing to engage users in friendly and lifelike conversations, making them feel like they are chatting with a real friend.

Display detected emotions on screen.
Users can easily see how their emotions are interpreted, with the help of creative pictures such as mic on, mic off and hourglass creating an interactive and entertaining aspect of the application.

Detect emotions accurately.
SER utilizes advanced machine learning algorithms to accurately identify and classify emotions in speech, providing a highly reliable assessment of the user's emotional state.
It takes into account various vocal cues, such as chroma, mel, MFCC and speech patterns, to ensure precise emotion detection, enhancing the overall entertainment experience.

Proper exception handling.
Our application is equipped with robust error handling mechanisms to gracefully handle unexpected situations, ensuring a smooth and uninterrupted user experience.

1.3 Purpose and Scope

1.3.1 Purpose

1. The primary purpose of this application is to create a genuinely interactive and emotionally intelligent chatbot that enhances user engagement by recognizing and responding to their emotions effectively.
2. By recognizing and displaying emotions from user queries, the application aims to provide a more personalized and empathetic conversational experience.
3. It serves as a tool for fostering emotional well-being, as it encourages users to express their feelings and thoughts, potentially leading to better self-awareness and emotional regulation.
4. It can be a fun and entertaining platform for users to engage with, as the chatbot's ability to mirror emotions adds a new dimension to entertainment and interactive storytelling.
5. In educational settings, the application can be used as a tool for teaching emotional intelligence and effective communication by showcasing the importance of understanding and responding to emotions in conversations.
6. The application can also be used for research and data collection, enabling the analysis of emotional trends and patterns in user interactions.
7. By promoting emotionally responsive AI, the application contributes to the development of AI systems that better understand and cater to human emotional needs.
8. It offers a safe space for users to share their feelings and receive non-judgmental responses, which can be particularly beneficial in situations where human interaction may be limited.
9. Overall, the primary purpose of this application is to harness the power of AI to create a more emotionally aware, empathetic, and engaging chatbot, facilitating deeper connections and personal growth while also providing entertainment and support to users.

1.3.2 Scope

1. The scope of this project encompasses the development of a sophisticated chatbot with emotion recognition capabilities.
2. It will include the integration of natural language processing and machine learning techniques to enable the chatbot to understand and respond to a wide range of user queries and emotions.
3. The chatbot's emotional recognition scope will cover a diverse set of emotional states such as happy,

angry, neutral, sad.
4. The application will employ real-time emotion detection, ensuring that user emotions are continuously monitored and displayed on the screen during conversations.
5. It will have a user-friendly interface that allows users to easily interact with the chatbot and observe their emotions reflected in real-time.
6. The chatbot will be designed to provide appropriate responses that match the detected emotions, creating a dynamic and emotionally responsive conversational experience.

Why it's important to "feel" all of your feelings
Emotions help us to communicate with others, such as when we feel sad and need some help.
https://wexnermedical.osu.edu/blog/why-its-important-to-feel-all-of-your-feelings
100%

Why it's important to "feel" all of your feelings | Ohio State ...
They also can help us to act quickly in important situations.
https://wexnermedical.osu.edu/blog/why-its-important-to-feel-all-of-your-feelings#:~:text=Emotions%20help%20us%20to%20communicate,jump%20back%20onto%20the%20curb.
100%

psychcentral.com › lib › why-are-feelings-importantWhy Are Emotions So Important? And How to Address Them - Psych...
May 6, 2022 · They can help you survive, grow, and connect with others. And they can guide your decisions, behaviors, and motivations. As babies, emotions are how you learn to communicate, even before you can...
https://psychcentral.com/lib/why-are-feelings-important
45%

Broad Mind - Strategies For Mental Health - X.com
They can help you survive, grow, and connect with others.
https://twitter.com/broad_mindsmh?lang=en
100%

Why Are Emotions So Important? And How to Address Them
As babies, emotions are how you learn to communicate, even before you can talk.
https://psychcentral.com/lib/why-are-feelings-important
100%

Neural Networks. Sci kit learn | by Robert Shaneyfelt - Medium
on an underlying Neural Network to perform the task of classification.
https://medium.com/illumination/neural-networks-f72e1cb67f5d#:~:text=MLPClassifier%20stands%20for%20Multi%2Dlayer,perform%20the%20task%20of%20classification.
100%

www.javatpoint.com › librosa-library-in-pythonLibrosa Library in Python - Javatpoint
Librosa is valuable Python music and sound investigation library that helps programming designers to fabricate applications for working with sound and music document designs utilizing Python. This Python bundle for music and sound examination is essentially utilized when we work with sound information, like in the music age (utilizing Lstm's ...
https://www.javatpoint.com/librosa-library-in-python
62%

Glossary

SER - Speech emotion recognition (SER) is the process of predicting human emotions from audio signals using artificial intelligence (AI) techniques.

Librosa - librosa is a python package for music and audio analysis. It provides the building blocks necessary to create music information retrieval systems.

MLPClassifier - The multilayer perceptron (MLP) is a feedforward artificial neural network model that maps input data sets to a set of appropriate outputs.

Voice features:

Chroma - The chroma feature is a descriptor, which represents the tonal content of a musical audio signal in a condensed form.

Mel - The mel scale is a scale of pitches judged by listeners to be equal in distance one from another.

Mel-frequency cepstral coefficients(MFCC) - MFCC is a feature extraction technique widely used in speech and audio processing. MFCCs are used to represent the spectral characteristics of sound in a way that is well-suited for various machine learning tasks, such as speech recognition and music analysis.

Appendix

This Project has mostly only been able to predict angry, sad and happy, Likewise other emotions were also removed while data preprocessing to achieve higher accuracy and avoid confusion. This result is justified by a survey on Speech Emotion Recognition conducted by Kunal Bhapkar1, Krati Patni2, Praddyumn Wadekar3, Shweta Pal4, Dr. Rubeena A. Khan5, Mahesh Shinde6. According to their study a Machine Learning model has only been able to classify three emotions i.e.Angry, Sad and Happy.

Lets look of the various other methods studied by them for Speech Emotion Recognition:

Sr no	Title	Summary
1	Speech based Emotion Recognition using Machine Learning[1]	Was only limited to classify three emotions i.e., Angry, Sad and Happy
2	Speech Emotion Recognition Based on Deep Belief Network[2]	Emotions such as shame and surprise cannot be identified, affecting the overall rate of recognition and average rate of recognition
3	An Enhanced Human Speech Emotion Recognition Using Hybrid of PRNN and KNN[3]	Six universal feelings are classified like neutral, sadness, happiness, angry, fear and surprise over the given input
4	Speech based human emotion Recognition using MFCC[4]	Only one feature extraction is used i.e., MFCC. Due to which, only three emotions are confirmed irrespective of the gender

5	Research on Emergency Parking Instruction Recognition Based on Speech Recognition and Speech Emotion Recognition[5]	If the system recognises the feeling of the parking education expression, the efficiency is degraded due to voice stopping guidance database utilized in this paper are recorded totally extraordinary surroundings
6	New Trends in Speech Emotion Recognition[6]	It obtained less accuracy. Changing dataset can prove as a solution to this problem
7	Emotion recognition from speech signal[7]	It just analyses characteristics of various four emotion states & does'nt include the classification of emotions
8	Speech Emotion Recognition Using Deep Learning Techniques: A Review[8]	The research work is done for different DNN techniques but no such implementation are done. It was just a theoretical concept
9	Improving Speech Emotion Recognition with Unsupervised Representation Learning on Unlabeled Speech[9]	Representations became similar to factors. Henceforth, in 2D projections, separate clusters were not found(bound to space limitations)
10	Speech emotion recognition with deep learning[10]	DNN had no understanding of the real sense of what the actor try to say. Neither it had any understanding of the speech, vibrations etc

Approved Project Proposal

By Rehmah Ahmed Batki TYCS-05

Title

Interactive problem solving chatbot with Speech Emotion Recognition(SER) Desktop application with Python. - SPEECH EMOTION RECOGNITION SOFTWARE

Introduction

SERS will be a desktop application that will be your personal bot friend, but this is not like any other chatbots or virtual assistants out there. SERS will be featured with speech conversing ability and to be able to read your mood from the tone of your voice! That's right, SERS will be a speech conversing friend bot that can detect emotion from speech and respond accordingly.

Objectives

On completion of this project I aim to achieve the following-

- Fostering skills in Application Development in Python.
- Getting good hands in building, training and implementing an AI Model.
- Gaining experience in Software Development, Deployment and Maintenance.

Scope

SERS's current scope is to provide the user with-

- An attractive and easy to use application for entertainment purpose.
- Pose as a friend, converse on simple topics and pick up on your mood from your voice.
- Display detected emotions on screen.
- Respond according to your mood.

Methodology

The Software Development Lifecycle Model that will be used for SERS is the Incremental Model as it best fits my softwares development process.

The incremental model suggests building 1 single functionality at a time via 4 stages making a single increment.

- Requirement Gathering
- Planning and Design
- Implementation and Testing
- Evaluation and Feedback

Each increment builds upon the previous increments until the final product is achieved.

For SERS, I have 4 major functionalities to work on, in the following order-

1. Chatbot Response Algorithm.
2. Speech Conversing Ability.
3. Building, Training and Implementing Speech Emotion Recognition Model.
4. Developing Desktop Application and generating EXE file.

Each of them will be planned, designed, coded, tested(Unit Testing) and reviewed before moving to the next one in the above given sequence.

After completion of a functionality it will be integrated with the previously built functionality and Integration testing will be performed. After the final functionality is coded and tested, System testing will be performed.

Tools and Technologies

1. SERS will be developed with Python Programming Language on JetBrains Pycharm IDE.
2. Model will be trained on Jupyter Notebook
3. Packages I will be using-
    a. Chatbot Response Algorithm: re, random.
    b. Speech Conversing Ability: speech_recognition, gtts, playsound, pyaudio.
    c. Speech Emotion Recognition Model: librosa, soundfile, numpy, scikit-learn,
    d. Android Application Development: customtkinter

Timeline

July:

10th - 21st July: Chatbot Response Algorithm + Speech Conversing Ability

Requirement Analysis, Design, Coding, Testing (Unit and Integration Testing).

July-August:

22nd July - 20th August: Speech Emotion Recognition Model

Requirement Analysis (sound features study), Designing, Coding (building & Training), Accuracy check, Implementing (in the main program), Testing (Unit and Integration Testing).

August:

21st - 31st August: Desktop Application Development

Requirement Analysis (user-friendly GUI), Design, Coding, Testing (Unit and System Testing).

September:

1st - 9th September: Review/Feedback, any changes, improvement or enhancement.

10th September: Deployment.

Resources

1) Software: Python, JetBrains IDE, Jupyter Notebook
2) System Requirements: 64 bit, 4gb ram, 500gb rom, intel core i3
3) Dataset: RAVDESS dataset; this is the Ryerson Audio-Visual Database of Emotional Speech and Song dataset.

Expected Outcome

- Visually pleasing and easy to use UI.
- Smooth and light application.
- Accurately evaluate emotion behind one's voice.
- Proper exception handling in any case.
- I myself, am expected to be well-versed in various terminologies related to sound features, AI model training, testing and application development.

References

[1] Article on Speech Emotion Recognition using Python | Sound Classification | Deep Learning at hackersrealm.net, uploaded in June 2022.

[2] Research Paper on “Speech based Emotion Recognition using Machine Learning” by Sukanya anil Kulkarni and her fellow mates at 3rd International Conference on Computing Methodologies and Communication (ICCMC) in 2019

[3] Research Paper on “Machine Learning Based Speech Emotions Recognition System” by Dr. Yogesh Kumar, Dr. Manish Mahajan in INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 8, ISSUE 07, JULY 2019

[4] Research Paper on “Speech Emotion Recognition System Using Machine Learning” by Husbaan I. Attar, Nilesh K. Kadole, Omkar G. Karanjekar, Devang R. Nagarkar, Prof. Sujeet and

More in the International Journal of Research Publication and Reviews, Vol 3, no 5, May 2022

[5] Research Paper - SpeechFormer++: A Hierarchical Efficient Framework for Paralinguistic Speech Processing by Weidong Chen, Xiaofen Xing, Xiangmin Xu, Jianxin Pang, and Lan Du, 27 Feb 2023

[6]Research Paper on TEMPORAL MODELING MATTERS: A NOVEL TEMPORAL EMOTIONAL MODELING APPROACH FOR SPEECH EMOTION RECOGNITION by Jiaxin Ye , Xin-cheng Wen , Yujie Wei1 , Yong Xu , Kunhong Liu4 , Hongming Shan, 10 March 2023.