EMOTION RECOGNITION USING SPEECH PROCESSING

A Project (Phase-I) Report

Submitted To

Chhattisgarh Swami Vivekanand Technical University
Bhilai, India

For

The Award of Degree

of

Bachelor of Technology

in

Computer Science & Engineering

By

Amit Prasad
Roll No.- 303302220109
En. No. – BJ4833
Semester 7th (CSE)

Aalind Shukla
Roll No.-303302220084
En. No. – BJ4792
Semester 7th (CSE)

Under the Guidance of
Mrs. Priyata Mishra
Asst. Professor
Department of Computer Science & Engineering
S.S.I.P.M.T, Raipur

Department of Computer Science & Engineering
Shri Shankaracharya Institute of Professional Management & Technology Raipur (C.G.)

Session: 2023 – 2024

DECLARATION BY THE CANDIDATE

We the undersigned solemnly declare that the project(phase-I) report entitled "***EMOTION RECOGNITION USING SPEECH PROCESSING***" is based our own work carried out during the course of our study under the supervision of **Mrs.Priyata Mishra .** We assert that the statements made and conclusions drawn are an outcome of the project work. We further declare that to the best of our knowledge and belief that the report does not contain any part of any work which has been submitted for the award of any other degree/diploma/certificate in this University/Deemed university of India or any other country.

(Signature of the Student)
Amit Prasad
Roll No.- 303302220109
En. No. – BJ4833
Semester 7th (CSE)

(Signature of the Student)
Aalind Shukla
Roll No.- 303302220084
En. No. – BJ4792
Semester 7th (CSE)

CERTIFICATE BY THE SUPERVISOR

This is to certify that the Major project report entitled **"*EMOTION RECOGNITION USING SPEECH PROCESSING*"** is a record of project work carried out under my guidance and supervision for the fulfillment of the award of degree of Bachelor of Technology in the faculty of Computer Science & Engineering of Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.) India.

To the best of my knowledge and belief the report

i) Embodies the work of the candidate himself

ii) Has duly been completed

iii) Fulfills the partial requirement of the ordinance relating to the B.E. degree of the University

iv) Is up to the desired standard both in respect of contents and language for being referred to the examiners.

(Signature of the Supervisor)

Mrs. Priyata Mishra

Assistant Professor, Dept of C.S.E.
S.S.I.P.M.T, Raipur (C.G.)

Forwarded to

Chhattisgarh Swami Vivekanand Technical University

Bhilai

(Signature of HOD)

Prof. Riju Bhattacharya

Dept. of Computer Science & Engineering
S.S.I.P.M.T, Raipur, C.G

(Signature of the Principal)

Dr. Alok Kumar Jain

S.S.I.P.M.T, Raipur, C.G

CERTIFICATE BY THE EXAMINERS

The project report entitled **"*EMOTION RECOGNITION USING SPEECH PROCESSING*"** "has been examined by the undersigned as a part of the examination of Bachelor of Technology in the faculty of Computer Science & Engineering of Chhattisgarh Swami Vivekanand Technical University, Bhilai.

Internal Examiner	**External Examiner**
Date:	**Date:**

ACKNOWLEDGEMENT

Working for this project has been a great experience for us. There were moments of anxiety, when we could not solve a problem for the several days. But we have enjoyed every bit of process and are thankful to all people associated with us during this period we convey our sincere thanks to our project guide **Mrs.Priyata Mishra** for providing me all sorts of facilities. His support and guidance helped us to carry out the project. We owe a great dept. of his gratitude for his constant advice, support, cooperation & encouragement throughout the project we would also like to express our deep gratitude to respected **Prof. Riju Bhattacharya (Head of Department)** for his ever helping and support. We also pay special thanks for his helpful solution and comments enriched by his experience, which improved our ideas for betterment of the project. We would also like to express our deep gratitude to respected **Dr. Alok Kumar Jain (Principal)** and college management for providing an educational ambience. It will be our pleasure to acknowledge, utmost cooperation and valuable suggestions from time to time given by our staff members of our department, to whom we owe our entire computer knowledge and also we would like to thank all those persons who have directly or indirectly helped us by providing books and computer peripherals and other necessary amenities which helped us in the development of this project which would otherwise have not been possible.

(Signature of the Student)
Amit Prasad
Roll No.- 303302220109
En. No. – BJ4833
Semester 7th (CSE)

(Signature of the Student)
Aalind Shukla
Roll No.- 303302220084
En. No. – BJ4792
Semester 7th (CSE)

ACKNOWLEDGEMENT –AICTE IDEA Lab

We have taken efforts in this project. However, it would not have been possible without the kind support and help of AICTE-IDEA Lab at SSIPMT, Raipur. We would like to extend our sincere thanks to all the gurus, mentors and support staff of Idea lab.

LIST OF ABBREVIATIONS

DNN	Deep neural networks
KNN	K-Nearest Neighbour
SVM	Support Vector Machine
ANN	Artificial neural networks
HMM	Hidden Markov Models
GMM	Gaussian mixture model
HCI	Human-computer interaction
MFCC	Mel Frequency Cepstral Coefficients
LPCC	Linear prediction cepstral coefficients
DBN	a deep belief network
RAVDESS	Ryerson Audio-Visual Database of Emotional Speech and Song

LIST OF FIGURES

ABSTRACT

Speech emotion recognition (SER) is the process of automatically identifying the emotional state of a speaker from their speech signal. It has gained significant interest in recent years due to its potential applications in human-computer interaction, customer service, healthcare, and education. This paper presents a comprehensive overview of SER with a focus on machine learning (ML) techniques.The paper begins by discussing the challenges of SER and the various emotional categories typically used in such systems. It then describes the three main stages of an ML-based SER system: data preprocessing, feature extraction, and classification. The ML algorithm used for this project MLP Classifier.

Keywords— Speech emotion recognition, Machine Learning, Deep Learning, Speech Features.

TABLE OF CONTENTS

CHAPTER-1

INTRODUCTION

1.1 INTRODUCTION

There are many ways of communication but the speech signal is one of the fastest and most natural methods of communications between humans. Therefore, the speech can be the fast and efficient method of interaction between human and the machine as well. Humans have the natural ability to use all their available senses for maximum awareness of the received message. Through all the available senses people actually sense the emotional state of their communication partner. The emotional detection is natural for humans but it is very difficult task for machine. Therefore, the purpose of emotion recognition system is to use emotion related knowledge in such a way that human machine communication will be improved.In this system, the quality of feature extraction directly affected the accuracy of speech emotion recognition. In the process of feature extraction, it usually took the whole emotion sentence as units for feature extracting, and extraction contents were four aspects of emotion speech, which were several acoustic characteristics of time construction, amplitude construction, fundamental frequency construction, and formant construction. Then contrast emotion speech with no emotion sentence from these four aspects, acquiring the law of emotional signal distribution,then classify emotion speech according to the law.Deep neural network (DNN) has unprecedented success in the field of speech recognition and image recognition; however, so far no research on deep neural network has been applied to speech emotion processing. We found that the DNN in speech emotion processing has a huge advantage. Therefore, this paper proposed a method to realize the emotional features automatically extracted from the audio using the librosa package in python. We used DNN to train a 5-layer-deep network to extract speech emotion features. It incorporates the speech emotion features of more consecutive frames, to build a high latitude characteristic, and uses softmax classifier layer to classify the emotional speech. The speech emotion recognition test accuracy reached 73.38% which is a high value compared to the other models of this size. Traditional machine learning methods are k-nearest neighbors (KNN), Hidden Markov Model(HMM) and Support Vector Machine (SVM), Artificial Neural Network (ANN), Gaussian Mixtures Model (GMM) etc to classify the emotions.The important issues in speech emotion recognition system are the signal processing unit in which appropriate features are extracted from available speech signal and another is a classifier which recognizes emotions from the speech signal. The average accuracy of the most of the classifiers for speaker independent system is less than that for the speaker dependent. Automatic emotion recognitions from the human speech are increasing now a day because it results in the better interactions between human and machine.

1.2 PROBLEM DEFINITION

The problem statement of emotional recognition using speech processing is to develop algorithms and techniques that can accurately identify and classify the emotions expressed inhuman speech. This involves analyzing various acoustic features of speech, such as pitch,volume, and tempo, and using machine learning techniques to identify patterns that correspondto different emotional states, such as happiness, sadness, anger, or fear. The ultimate goal is to develop robust and reliable models that can automatically recognize emotions in real- time applications, such as customer service interactions, virtual assistants, or mental health assessments. This field has important implications for improving human-computer interaction,mental health diagnosis, and communication research.

1.2.1 OBJECTIVE

Human speech is the most natural way to express ourselves. We use it everywhere from calls, emails, meetings, discussions etc. As emotions play a vital role in communication, the detection and analysis of the same is of vital importance in today's digital world of remote communication. This project can be defined as a collection of methodologies that process and classify speech signals to detect emotions in them. The objective is to detect the emotions of a person or speaker and to implement a Deep Neural Network (DNN) model to create the application.

CHAPTER-2

LITERATURE REVIEW

2.1 LITERATURE REVIEW

[1] Szegedy, Christian & Toshev, Alexander & Erhan, Dumitru. (2013). Deep Neural Networks for Object Detection. 1-9.

Deep Neural Networks (DNNs) have recently shown outstanding performance on image classification tasks [14]. In this paper we go one step further and address the problem of object detection using DNNs, that is not only classifying but also precisely localizing objects of various classes. We present a simple and yet powerful formulation of object detection as a regression problem to object bounding box masks. We define a multi-scale inference procedure which is able to produce high-resolution object detections at a low cost by a few network applications. State-of-the-art performance of the approach is shown on Pascal VOC.

Summary: This journal discusses about the Deep Neural Networks theory and object detectionusing DNN

[2] Benk, Sal & Elmir, Youssef & Dennai, Abdeslem. (2019). A Study on Automatic Speech Recognition. 10. 77-85. 10.6025/jitr/2019/10/3/77-85.

Speech is an easy and usable technique of communication between humans, but nowadays humans are not limited to connecting to each other but even to the different machines in ourlives. The most important is the computer. So, this communication technique can be used between computers and humans. This interaction is done through interfaces, this area called Human Computer Interaction (HCI). This paper gives an overview of the main definitions ofAutomatic Speech Recognition (ASR) which is an important domain of artificial intelligence and which should be taken into account during any related research (Type of speech,vocabulary size... etc.). It also gives a summary of important research relevant to speech processing in the few last years, with a general idea of our proposal that could be considered as a contribution in this area of research and by giving a conclusion referring to certain enhancements that could be in the future works.

Summary: This article helps us in understanding and using the speech recognition by machines which improves Human Computer Interactions and is also useful in our project.

[3] Ashish B. Ingale & D. S. Chaudhari (2012). Speech Emotion Recognition. International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2 Issue-1, March 2012

In human machine interface application, emotion recognition from the speech signal has been research topic since many years. To identify the emotions from the speech signal, many systems have been developed. In this paper speech emotion recognition based on the previous technologies which uses different classifiers for the emotion recognition is reviewed. The classifiers are used to differentiate emotions such as anger, happiness, sadness, surprise, neutral state, etc. The database for the speech emotion recognition system is the emotional speech samples and the features extracted from these speech samples are the energy, pitch, linear prediction cepstrum coefficient (LPCC), Mel frequency cepstrum coefficient (MFCC). The classification performance is based on extracted features. Inference about the performance and limitation of speech emotion recognition system based on the different classifiers are also discussed.

Summary: In this paper, we learn the importance and the need of a different features in any audio or speech including mfcc, mel and other features which are used in our application for the purpose of predicting the emotions based on audio.

[4] Chenchen Huang, Wei Gong, Wenlong Fu, Dongyu Feng, "A Research of Speech Emotion Recognition Based on Deep Belief Network and SVM", *Mathematical Problems in Engineering*, vol. 2014, Article ID 749604, 7 pages, 2014. https://doi.org/10.1155/2014/749604

Feature extraction is a very important part in speech emotion recognition, and in allusion to feature extraction in speech emotion recognition problems, this paper proposed a new method of feature extraction, using DBNs in DNN to extract emotional features in speech signal automatically. By training a 5 layers depth DBNs, to extract speech emotion feature and incorporate multiple consecutive frames to form a high dimensional feature. The features after training in DBNs were the input of nonlinear SVM classifier, and finally speech emotion recognition multiple classifier system was achieved. The speech emotion recognition rate of the system reached 86.5%, which was 7% higher than the original method.

CHAPTER-3

METHODOLOGY

3.1 METHODOLOGY

There are many ways of communication but the speech signal is one of the fastest and most natural methods of communications between humans. Therefore, the speech can be the fast and efficient method of interaction between human and the machine as well. Humans have the natural ability to use all their available senses for maximum awareness of the received message. Through all the available senses people actually sense the emotional state of their communication partner. The emotional detection is natural for humans but it is very difficult task for machine. Therefore, the purpose of emotion recognition system is to use emotion related knowledge in such a way that human machine communication will be improved.

In this system, the quality of feature extraction directly affected the accuracy of speech emotion recognition. In the process of feature extraction, it usually took the whole emotion sentence as units for feature extracting, and extraction contents were four aspects of emotion speech, which were several acoustic characteristics of time construction, amplitude construction, fundamental frequency construction, and formant construction. Then contrast emotion speech with no emotion sentence from these four aspects, acquiring the law of emotional signal distribution, then classify emotion speech according to the law.

Deep neural network (DNN) has unprecedented success in the field of speech recognition and image recognition; however, so far no research on deep neural network has been applied to speech emotion processing. We found that the DNN in speech emotion processing has a huge advantage. Therefore, this paper proposed a method to realize the emotional features automatically extracted from the audio using the librosa package in python. We used DNN to train a 5-layer-deep network to extract speech emotion features. It incorporates the speech emotion features of more consecutive frames, to build a high latitude characteristic, and uses softmax classifier layer to classify the emotional speech. The speech emotion recognition test accuracy reached 73.38% which is a high value compared to the other models of this size.

We propose a modified speech emotion recognition method which uses deep neural networks for training. The method uses Mel-frequency cepstral coefficients (MFCC), Chromogram, Mel scaled spectrogram in conjunction with Spectral contrast and Tonal Centroid features to extract details about an audio file. The features are used to train DNN model in a 5 layer deep neural network. The dataset used here is the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS). We have only chosen the speech part which consists of 24 actors (gender

balanced) with 1440 audio files. The model classifies the speech audio in 8 different emotions namely neutral, calm, happy, sad, angry, fearful, disgust, surprised.

3.2 EXISTING METHOD

In most of the currently available systems, the model used for emotion recognition uses traditional Machine learning algorithms like Support Vector Machines (SVM), K-Nearest neighbors (KNN) etc. The accuracies of these models are low. However, there are other Deep learning models as well but they are generally trained using large datasets which takes a lot of time and hence are very complex models.

DISADVANTAGES:

Lower accuracy.

High computational complexity.

Requires high performance hardware to use the application.

3.3 Related works

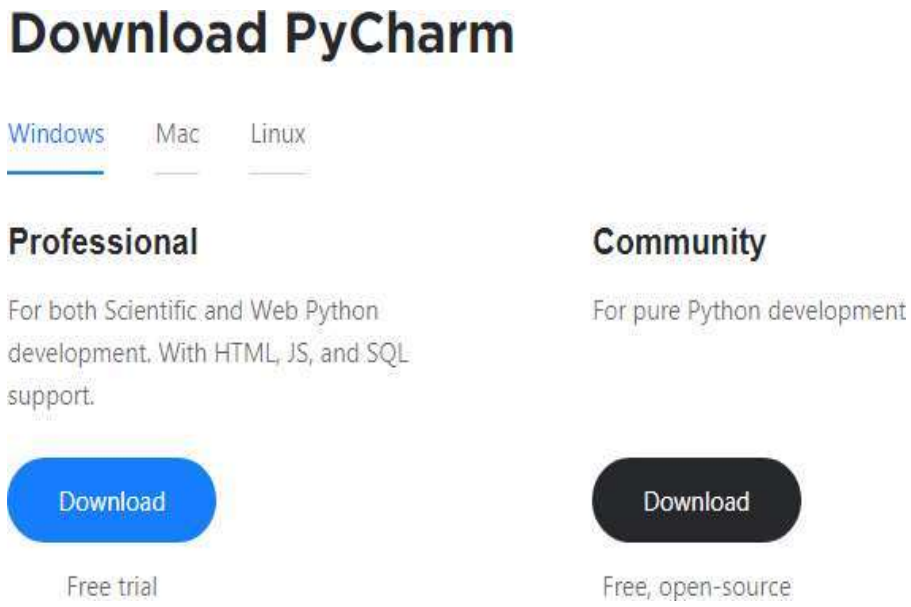
3.3.1 Installing Python

1. To download and install Python visit the official website of Python https://www.python.org/downloads/ and choose your version

2. Once the download is complete, run the exe for install Python. Now click on Install Now.
3. You can see Python installing at this point.
4. When it finishes, you can see a screen that says the Setup was successful. Now click on "Close".

3.3.2 Installing PyCharm:

1. To download PyCharm visit the website https://www.jetbrains.com/pycharm/download/ and Click the "DOWNLOAD" link under the Community Section.



2. Once the download is complete, run the exe for install PyCharm. The setup wizard should have started. Click “Next”.
3. On the next screen, Change the installation path if required. Click “Next”.
4. On the next screen, you can create a desktop shortcut if you want and click on “Next”.
5. Choose the start menu folder. Keep selected JetBrains and click on “Install”.
6. Wait for the installation to finish.

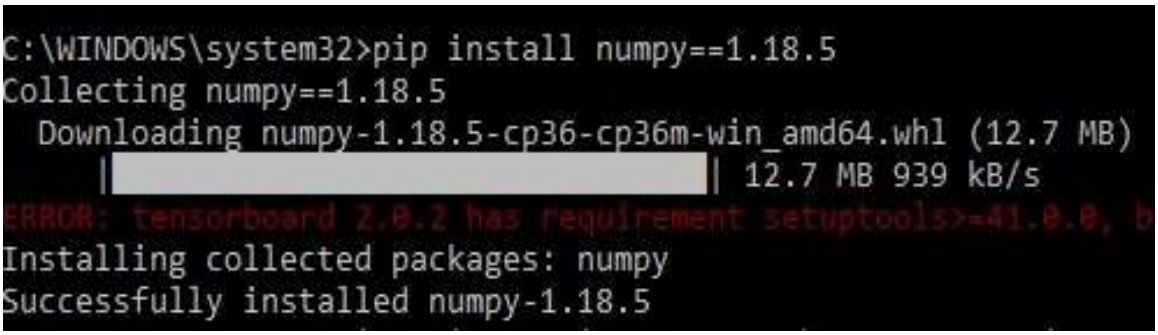
7. Once installation finished, you should receive a message screen that PyCharm is installed. If you want to go ahead and run it, click the “Run PyCharm Community Edition” box first and click “Finish”
8. After you click on "Finish," the Following screen will appear.

You need to install some packages to execute your project in a proper way.

9. Open the command prompt/ anaconda prompt or terminal as administrator.

10. The prompt will get open, with specified path, type “pip install package name” which you want to install (like numpy, pandas, seaborn, scikit-learn, matplotlib.pyplot)

Ex: pip install numpy



3.4 MODULES

1) Upload:

Upload the dataset of audio (.wav files) to be read using librosa library.

2) View:

Uploaded dataset can be viewed.

3) Pre-processing:

Data Pre-processing is a technique that is used to convert the raw data into a clean data set. Cleaning the data refers to removing the null values, filling the null values with meaningful value, removing duplicate values, removing outliers, removing unwanted attributes. If dataset contains any categorical records means convert those categorical variables to numerical values.

4) Identifying Features:

The extracted features are Mel-frequency cepstral coefficients (MFCC), Chromogram, Mel scaled spectrogram in conjunction with Spectral contrast and Tonal Centroid features.

5) Train and Test Split:

We split our dataset of 1440 audio files in 2 parts, training data with 1008 audio files and testing data with 432 audio files. Here 70% of the data is taken for the training dataset.

6) Building the model:

To understand the audio and predict emotions, we are proposing a Deep learning-based method Deep learning can provide increased accuracy and decrease in computational power.We will use

Deep Neural Networks (DNN) to create the model. Deep Neural Network (DNN) is widely used in deep learning to train models for tasks which traditional machine learning algorithms cannot do or is hard to doThe model is created using 5 layers of neural networks.We have used dropouts to minimize the problem of overfitting. In order to classify the audio to the emotions, we are using softmax in the outermost layer of our DNN model. Softmax takes in a vector of numbers and converts them to probabilities which are then used for image generating results. Softmax converts logits into probabilities by taking the exponents from every output and then normalize each of these numbers by the sum of such exponents, such that the entire output vector adds up to one.

8) Prediction:

An audio is uploaded by the user (which includes speech of a person), and the model is used to predict the emotion of the speaker in the audio.

9) User Interface:

A Flask architecture based web application is developed to use the model which is used to provide access to the application to predict the emotions.

3.5 Technologies used

Deep Learning – python library, used for implementation algorithm.
Flask , pycharm are used.

3.6 SYSTEM DESIGN

3.6.1 BLOCK DIAGRAM

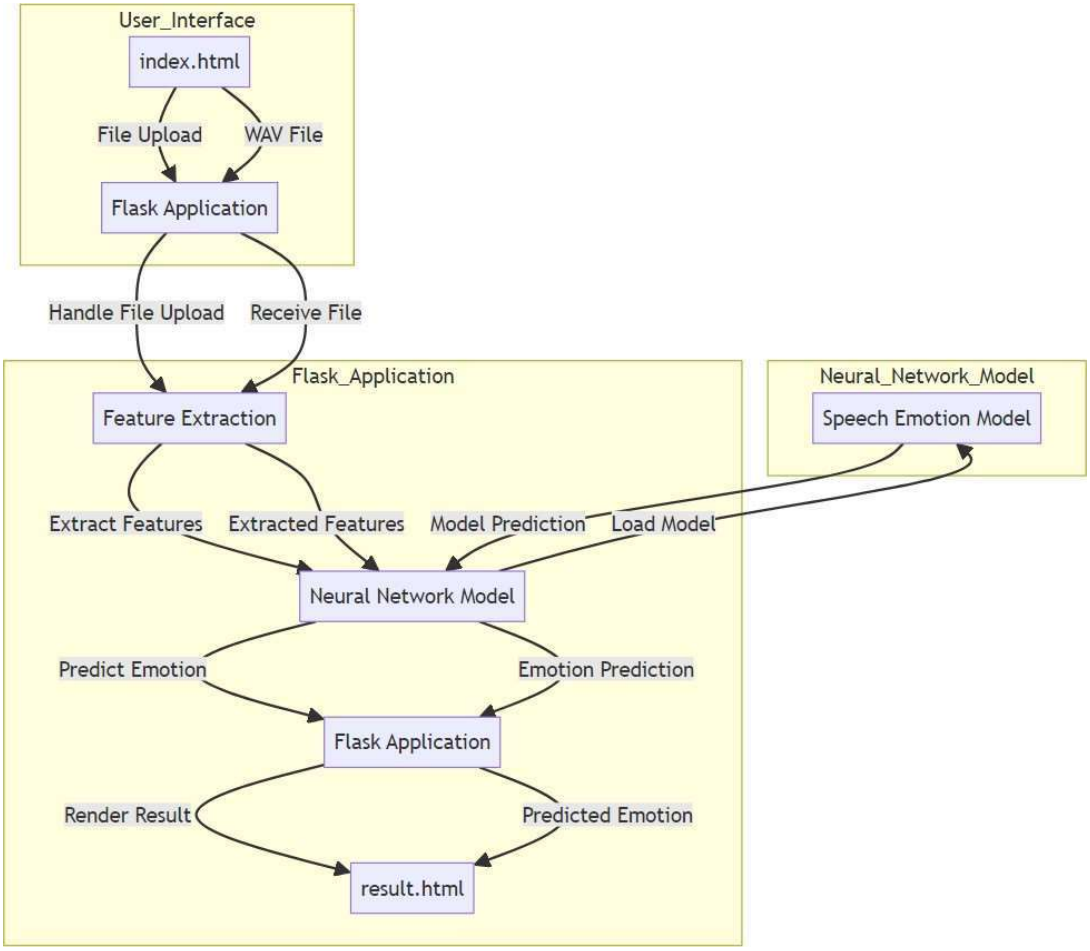


Fig 3.6.1 block diagram

3.6.2 USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

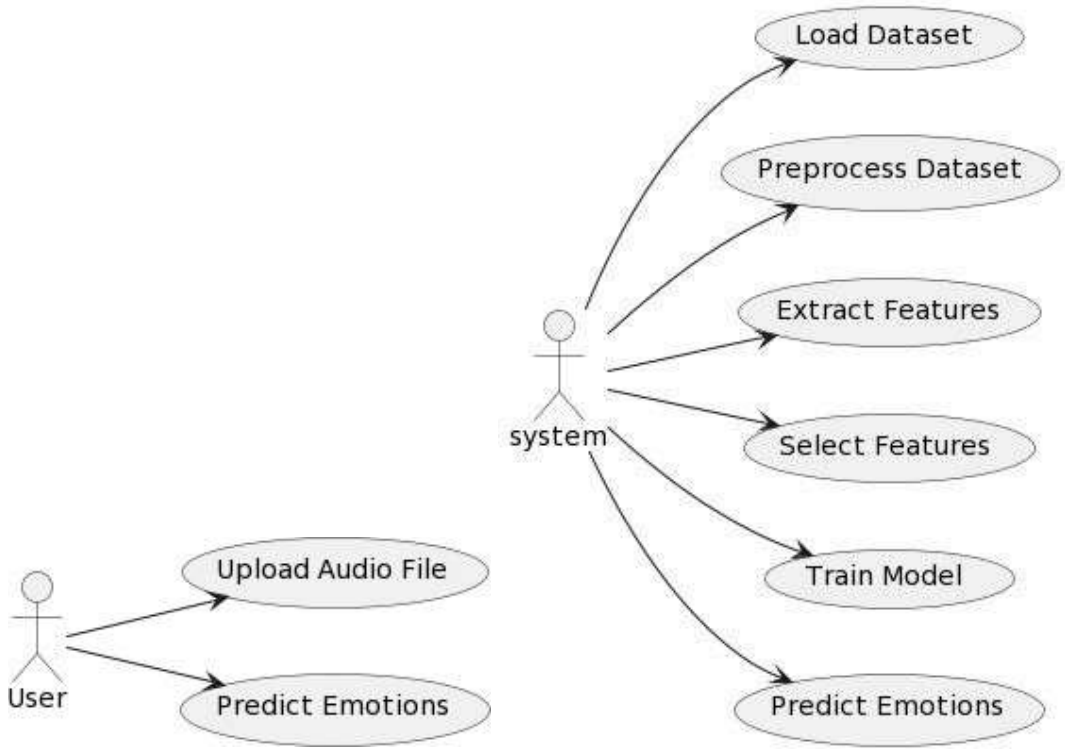


Fig 3.6.2 use case diagram.

3.6.3 CLASS DIAGRAM:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

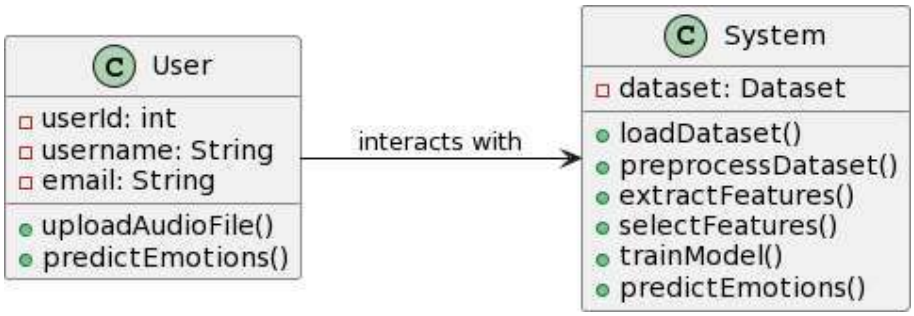


Fig 3.6.3 Class diagram

3.6.4 SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

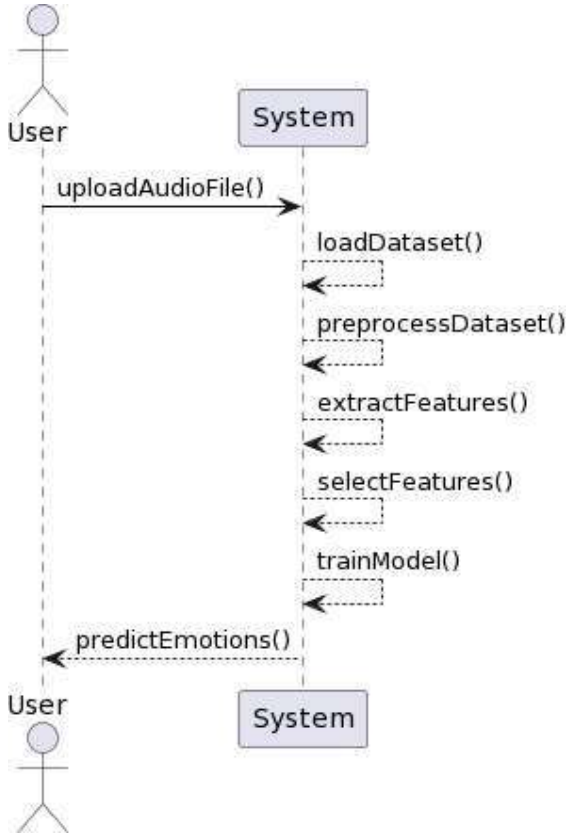


Fig 3.6.4 Sequence diagram

3.6.5 ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

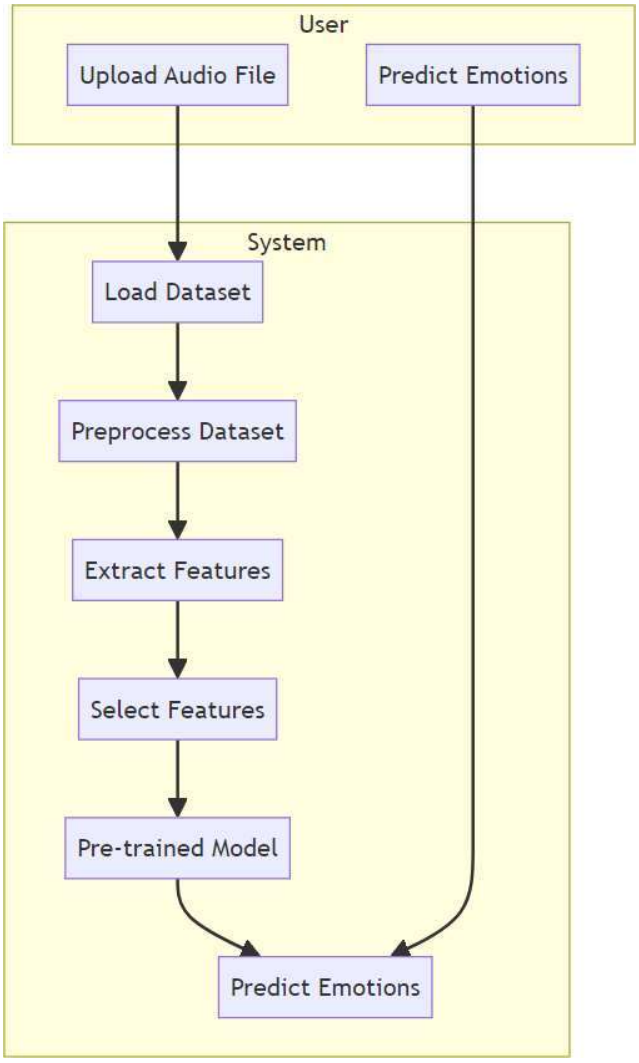


Fig 3.6.5 Activity Diagram

3.6.6 ER DIAGRAM:

An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set.

An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram

shows the complete logical structure of a database. Let's have a look at a simple ER diagram to understand this concept.

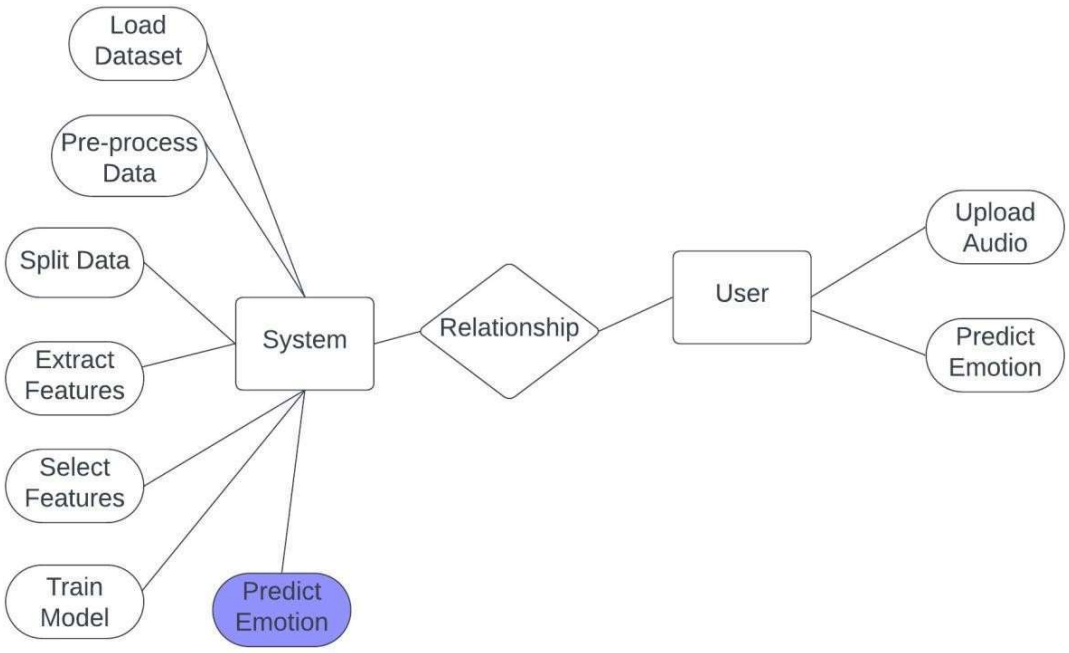


Fig 3.6.6 ER diagram

3.6.7 DFD DIAGRAM:

A Data Flow Diagram (DFD) is a traditional way to visualize the information flows within a system. A neat and clear DFD can depict a good amount of the system requirements graphically. It can be manual, automated, or a combination of both. It shows how information enters and leaves the system, what changes the information and where information is stored. The purpose of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communications tool between a systems analyst and any person who plays a part in the system that acts as the starting point for redesigning a system.

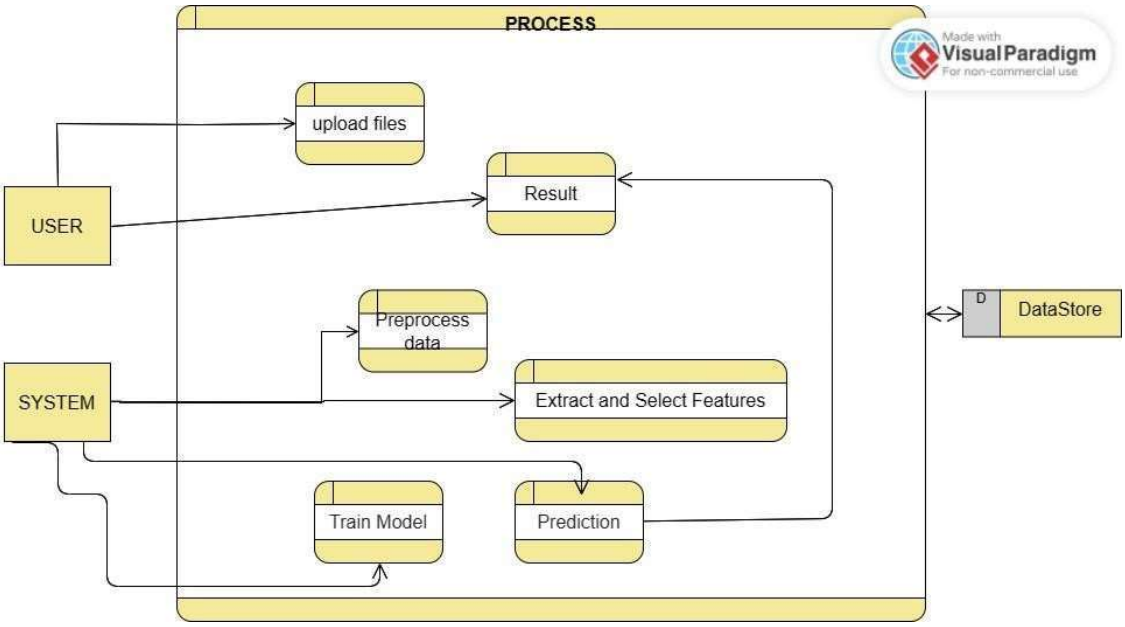


Fig 3.6.7 DFD diagram

3.6.8 Hardware and Software Requirement

3.6.8.1 Hardware Requirement

Processor	- I3/Intel Processor
RAM	- 4GB (min)
Hard Disk	- 128 GB
Key Board	- Standard Windows Keyboard
Mouse	- Two or Three Button Mouse
Monitor	- Any

3.6.8.2 Software Requirement

Operating System	: Windows 7+
Server side Script	: Python 3.6+
IDE	: PyCharm
Libraries Used	: Pandas, Numpy, Keras, Tensorflow, Librosa, OpenCV, Flask, Pickle.
Dataset	: RAVDESS speech dataset.

CHAPTER-4

RESULT

4.1 Output

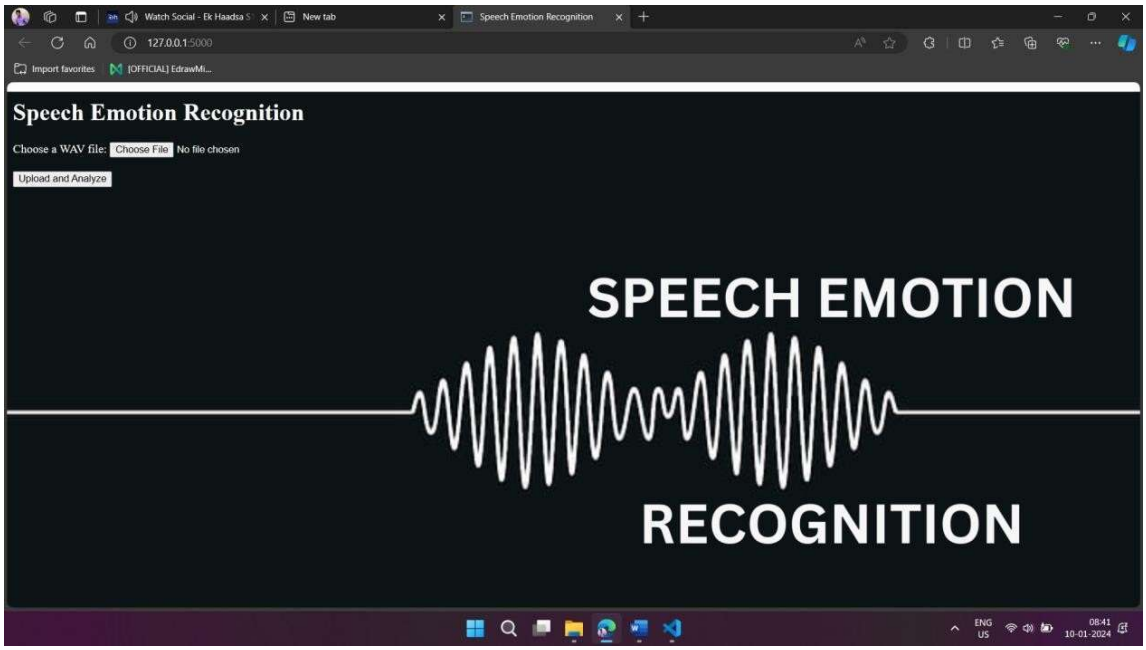


Fig 4.1 website.

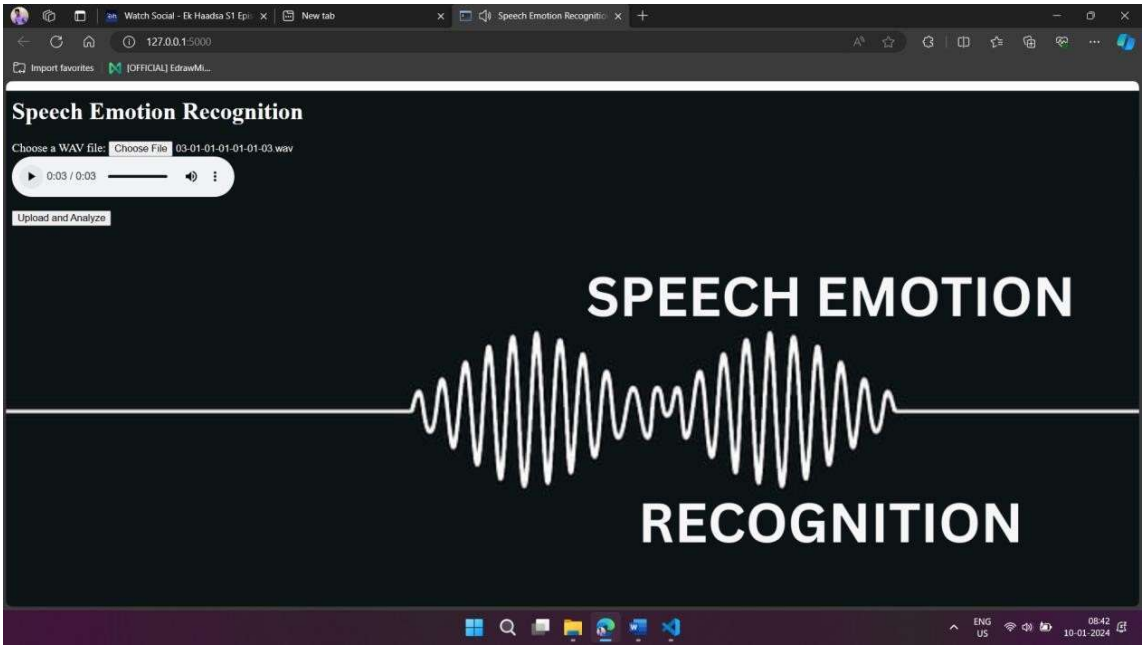


Fig 4.2 Audio Upload Tab.

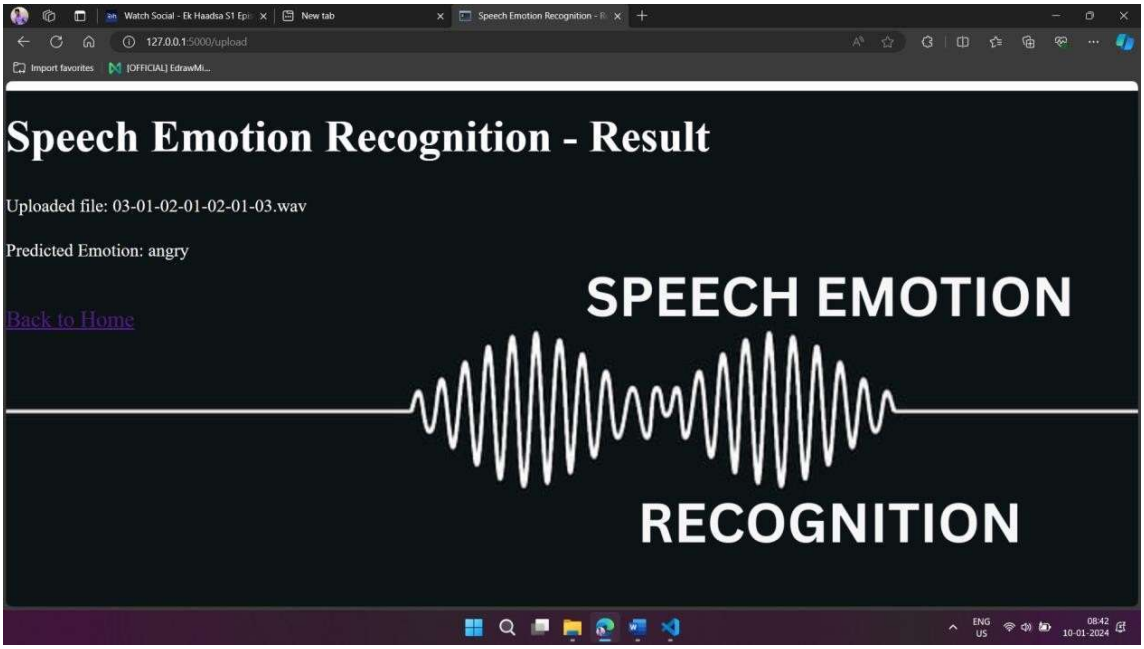


Fig 4.3 output result

4.2 CODES

4.2.1 Flask App

```
from flask import Flask, render_template, request
import os
from werkzeug.utils import secure_filename
import librosa
import numpy as np
from keras.models import load_model

app = Flask(__name__)

# Constants for the model and file upload
MODEL_PATH = "C:\\Users\\Aalin\\model\\hello.h5"
app.config['STATIC_FOLDER'] = 'static'
app.config['UPLOAD_FOLDER'] = 'uploads'
UPLOAD_FOLDER = 'uploads'
ALLOWED_EXTENSIONS = {'wav'}

EMOTIONS = ['neutral', 'calm', 'happy', 'sad', 'angry', 'fearful'
, 'disgusted', 'surprised']

app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER

# Load the pre-trained model
model = load_model(MODEL_PATH)

# ... (other functions and routes)

# Function to predict emotion from the features
def predict_emotion(features):
    # Predict using the pre-trained model
    prediction = model.predict(features.reshape(1, -1))

    # Get the index of the emotion with the highest probability
    predicted_class_index = np.argmax(prediction)

    # Get the name of the predicted emotion
    predicted_emotion_name = EMOTIONS[predicted_class_index]

    return predicted_emotion_name

app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER

# Load the pre-trained model
model = load_model(MODEL_PATH)

# Function to check if the file has a valid extension
def allowed_file(filename):
    return '.' in filename and filename.rsplit('.', 1)[1
].lower() in ALLOWED_EXTENSIONS
```

```
# Function to extract features from an audio file
def extract_features(file_path):
    X, sample_rate = librosa.load(file_path)
    # Extracting features (Placeholder, add actual code)
    mfccs = np.mean(librosa.feature.mfcc(y=X, sr=sample_rate,
n_mfcc=40).T, axis=0)
    chroma = np.mean(librosa.feature.chroma_stft(S=librosa.stft(X
), sr=sample_rate).T, axis=0)
    mel = np.mean(librosa.feature.melspectrogram(y=X, sr=
sample_rate).T, axis=0)
    contrast = np.mean(librosa.feature.spectral_contrast(S=
librosa.stft(X), sr=sample_rate).T, axis=0)
    tonnetz = np.mean(librosa.feature.tonnetz(y=librosa.effects.
harmonic(X), sr=sample_rate).T, axis=0)
    return np.hstack([mfccs, chroma, mel, contrast, tonnetz])

# Function to predict emotion from the features
# Function to predict emotion from the features
def predict_emotion(features):
    # Predict using the pre-trained model
    prediction = model.predict(features.reshape(1, -1))

    # Get the index of the emotion with the highest probability
    predicted_class_index = np.argmax(prediction)

    # Get the name of the predicted emotion
    predicted_emotion_name = EMOTIONS[predicted_class_index]

    print("Predicted Emotion:", predicted_emotion_name)
# Add this line

    return predicted_emotion_name
```

```
# Route for the home page
@app.route('/')
def home():
    return render_template('index.html')

# Route to handle the file upload
@app.route('/upload', methods=['POST'])
def upload_file():
    if 'file' not in request.files:
        return render_template('index.html', error='No file part'
)

    file = request.files['file']

    if file.filename == '':
        return render_template('index.html', error=
'No selected file')

    if file and allowed_file(file.filename):
        filename = secure_filename(file.filename)
        file_path = os.path.join(os.path.dirname(os.path.abspath(
__file__)), filename)
        file.save(file_path)

        # Extract features
        features = extract_features(file_path)

        # Predict emotion
        predicted_emotion = predict_emotion(features)

        return render_template('result.html', filename=filename,
emotion=predicted_emotion)

    return render_template('index.html', error=
'Invalid file format')

if __name__ == '__main__':
    app.run(debug=True)
```

4.2.2 SPEECH EXTRACT

```
#Import libraries
import glob
import os
import librosa
import numpy as np
from keras.models import Sequential
from keras.layers import Dense, Activation
from keras.layers import Dropout
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, accuracy_score
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from tensorflow.keras.callbacks import
 ReduceLROnPlateau, ModelCheckpoint


#Extract features
def extract_features(file_name):
    X, sample_rate = librosa.load(file_name)
    #Short time fourier transformation
    stft = np.abs(librosa.stft(X))
    #Mel Frequency Cepstra coeff (40 vectors)
    mfccs=np.mean(librosa.feature.mfcc(y=X, sr=sample_rate,
n_mfcc=40).T, axis=0)
    #Chromogram or power spectrum (12 vectors)
    chroma=np.mean(librosa.feature.chroma_stft(S=stft, sr=
sample_rate).T, axis=0)
    #mel scaled spectogram (128 vectors)
    mel=np.mean(librosa.feature.melspectrogram(y=X, sr=
sample_rate).T, axis=0)
    # Spectral contrast (7 vectors)
    contrast=np.mean(librosa.feature.spectral_contrast(S=stft, sr
=sample_rate).T, axis=0)
    #tonal centroid features (6 vectors)
    tonnetz=np.mean(librosa.feature.tonnetz(y=librosa.effects.
harmonic(X),sr=sample_rate).T, axis=0)
    return mfccs, chroma, mel, contrast, tonnetz
```

```
#parsing audio files
def parse_audio_files(parent_dir, sub_dirs, file_ext="*.wav"):
    features, labels = np.empty((0,193)), np.empty(0)
    for label, sub_dir in enumerate(sub_dirs):
        for fn in glob.glob(os.path.join(parent_dir, sub_dir,
file_ext)):
            try:
                mfccs, chroma, mel, contrast, tonnetz=
extract_features(fn)
            except Exception as e:
                print("Error encountered while parsing file: ",
fn)
                continue
            ext_features = np.hstack([mfccs, chroma, mel,
contrast, tonnetz])
            features = np.vstack([features, ext_features])
            labels = np.append(labels, fn.split("\\")[8].split(
"-")[2])
    return np.array(features), np.array(labels, dtype = np.int)

#fn=glob.glob(os.path.join(main_dir, sub_dir[0], "*.wav"))[0]

#One-Hot Encoding the multi class labels
def one_hot_encode(labels):
    n_labels = len(labels)
    n_unique_labels = len(np.unique(labels))
    one_hot_encode = np.zeros((n_labels, n_unique_labels + 1))
    one_hot_encode[np.arange(n_labels), labels] = 1
    one_hot_encode=np.delete(one_hot_encode, 0, axis=1)
    return one_hot_encode

#Extracting features in X
#Storing labels in y
main_dir = r'C:\Users\Aalin\OneDrive\Desktop\m
ajor project p1\Audio_Speech_Actors_01-24'
sub_dir = os.listdir(main_dir)
print("\nCollecting features and labels.")
print("\nThis will take some time.")
features, labels = parse_audio_files(main_dir, sub_dir)
```

```
#parsing audio files
def parse_audio_files(parent_dir, sub_dirs, file_ext="*.wav"):
    features, labels = np.empty((0, 193)), np.empty(0)
    total_files = 0

    for label, sub_dir in enumerate(sub_dirs):
        current_dir = os.path.join(parent_dir, sub_dir)
        file_list = glob.glob(os.path.join(current_dir, file_ext
))
        total_files += len(file_list)

        for idx, fn in enumerate(file_list):
            try:
                mfccs, chroma, mel, contrast, tonnetz =
extract_features(fn)
            except Exception as e:
                print("Error encountered while parsing file: ",
fn)
                continue

            ext_features = np.hstack([mfccs, chroma, mel,
contrast, tonnetz])
            features = np.vstack([features, ext_features])
            labels = np.append(labels, fn.split("\\")[8].split(
"-")[2])

            # Print progress
            print(f"\rProcessed {idx+1}/{len(file_list)}
 files in {sub_dir} - Total Progress: {len(features)}/{
total_files}", end="")
features, labels = parse_audio_files(main_dir, sub_dir)
print("\nCompleted")

#save features
np.save('X', features)
#one hot encode labels
labels = one_hot_encode(labels)
np.save('y', labels)

emotions=['neutral','calm','happy','sad','angry','fearful',
'disgused','surprised']
```

4.2.3 SPEECH TRAIN

```
#Import libraries
import glob
import os
import librosa
import numpy as np
from keras.models import Sequential
from keras.layers import Dense, Activation
from keras.layers import Dropout
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, accuracy_score
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from tensorflow.keras.callbacks import
 ReduceLROnPlateau, ModelCheckpoint, EarlyStopping
import keras
from keras import regularizers

#Loading features and labels
X=np.load('X.npy') #features
y=np.load('y.npy') #labels

#Splitting the dataset
train_X, test_X, train_y, test_y = train_test_split(X, y,
test_size= 0.3,random_state=42)

def get_network():
    input_shape = (193,)
    num_classes = 8
    keras.backend.clear_session()
```

```
    model = keras.models.Sequential()
     model.add(keras.layers.Dense(1024, activation="relu",
input_shape=input_shape))
     model.add(Dropout(0.5))
     model.add(keras.layers.Dense(512, activation="relu",
input_shape=input_shape))
     model.add(keras.layers.Dense(256, activation="relu",
input_shape=input_shape))
     model.add(keras.layers.Dense(128, activation="relu",
input_shape=input_shape))
     model.add(keras.layers.Dense(num_classes, activation =
"softmax"))
     model.compile(optimizer='adam',
         loss='categorical_crossentropy',
         metrics=["accuracy"])
     return model

model = get_network()

# Model Training
lr_reduce = ReduceLROnPlateau(monitor='val_accuracy', factor=0.9
, patience=20, min_lr=0.000001)
# Please change the model name accordingly.
mcp_save = ModelCheckpoint('model/hello.h5', save_best_only=True
, monitor='val_accuracy', mode='max')

#callbacks = [EarlyStopping(monitor='val_loss', mode='min', patie
nce=20), mcp_save, lr_reduce]

history=model.fit(train_X, train_y, epochs = 700, batch_size = 24
, validation_data=(test_X, test_y), callbacks=[mcp_save,
lr_reduce])

#l, a = model.evaluate(x_test, y_test, verbose = 0)

#Plots
# Plotting the Train Valid Loss Graph

plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.title('model accuracy: '+str(max(history.history[
'val_accuracy'])))
plt.ylabel('accuracy')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
```

CHAPTER-5

CONCLUSION

The proposed scheme presented an approach to recognize the emotion from the human speech. This approach has been implemented by the using the neural networks. We have successfully developed a deep learning model using the deep neural network architecture to predict the emotions of the speaker in an audio. We have famed our project in a web based application using the Flask architecture. The UI also includes user registration system. We were able to get a test accuracy of 73.4% using the trained model. Please note that emotion prediction is subjective and the emotions rated by a person for the same audio can differ from person to person. This is also the reason why the algorithm which is trained on human rated emotions can generate erratic results sometimes. The model was trained of RAVDESS dataset, so the accent of the speaker can also lead to erratic results as the model is only trained on North American accent database.

REFERENCES

[1] Szegedy, Christian & Toshev, Alexander & Erhan, Dumitru. (2013). Deep Neural Networks for Object Detection. 1-9.

[2] Benk, Sal & Elmir, Youssef & Dennai, Abdeslem. (2019). A Study on Automatic Speech Recognition. 10. 77-85. 10.6025/jitr/2019/10/3/77-85.

[3] Ashish B. Ingale & D. S. Chaudhari (2012). Speech Emotion Recognition. International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2 Issue-1, March 2012

[4] Chenchen Huang, Wei Gong, Wenlong Fu, Dongyu Feng, "A Research of Speech Emotion Recognition Based on Deep Belief Network and SVM", *Mathematical Problems in Engineering*, vol. 2014, Article ID 749604, 7 pages, 2014. https://doi.org/10.1155/2014/749604

[5] Livingstone SR, Russo FA (2018) The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. PLoS ONE 13(5): e0196391.

[6] M. E. Ayadi, M. S. Kamel, F. Karray, “Survey on Speech Emotion Recognition: Features, Classification Schemes, and Databases”, Pattern Recognition 44, PP.572-587, 2011.

[7] I. Chiriacescu, “Automatic Emotion Analysis Based On Speech”, M.Sc. THESIS Delft University of Technology, 2009.

[8] T. Vogt, E. Andre and J. Wagner, “Automatic Recognition of Emotions from Speech: A review of the literature and recommendations for practical realization”, LNCS 4868, PP.75-91, 2008.

[9] S. Emerich, E. Lupu, A. Apatean, “Emotions Recognitions by Speech and Facial Expressions Analysis”, 17th European Signal Processing Conference, 2009.

[10] P.Shen, Z. Changjun, X. Chen, “Automatic Speech Emotion Recognition Using Support Vector Machine”, International Conference On Electronic And Mechanical Engineering And Information Technology, 2011.