REAL TIME IDENTIFICATION OF HUMAN EMOTION USING ACOUSTIC FEATURES

by

DHARNI S S	**2016103022**
NEHA B	**2016103555**
SWETHA M	**2016103601**

A project report submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ANNA UNIVERSITY, CHENNAI – 25

APRIL 2020

BONAFIDE CERTIFICATE

Certified that this project report titled **REAL TIME IDENTIFICATION OF HUMAN EMOTION USING ACOUSTIC FEATURES** is the *bonafide* work of **DHARNI S S (2016103022)**, **NEHA B (2016103555)** and **SWETHA M (2016103601)** who carried out the project work under my supervision, for the fulfillment of the requirements for the award of the degree of Bachelor of Engineering in Computer Science and Engineering. Certified further that to the best of my knowledge, the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or an award was conferred on an earlier occasion on these or any other candidates.

Place: Chennai
Date:

Dr. Shanthi A P
Professor
Department of Computer Science and Engineering
Anna University, Chennai – 25

COUNTERSIGNED

Head of the Department,
Department of Computer Science and Engineering,
Anna University Chennai,
Chennai – 600025

ACKNOWLEDGEMENT

We express our deep gratitude to our guide, **Dr. A P Shanthi**, Professor, Department of Computer Science and Engineering, for guiding us through every phase of the project. She has encouraged us to think on our own and give form to our thoughts. We owe her for harnessing our potential and bringing out the best in us. Without her immense support through every support of the way, we could never have it to this extent.

We are extremely grateful to **Dr. S. Valli**, Professor and Head of the Department of Computer Science and Engineering, Anna University, Chennai - 25, for extending the facilities of the Department towards our project and for her unstinting support.

We express our thanks to the panel of reviewers **Prof. Arul Siromoney**, Professor, Department of Computer Science and Engineering, **Dr. Sudha S**, Associate Professor, Department of Computer Science and Engineering and **Dr. D. Shiloah Elizabeth**, Assistant Professor, Department of Computer Science and Engineering, for their valuable suggestions and critical reviews throughout the course of our project.

We thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.

Dharni S S **Neha B** **Swetha M**

ABSTRACT

A Speech emotion recognition system focuses on recognizing the state of emotion of a human being's voice. Such emotions are subjective to people and annotating an audio recording is challenging. SER technology has improved the areas in Human computer interfaces (HCI) such as in gaming, eLearning, voice search, etc.

Our proposed system focuses on developing a real time system which recognises the human emotion using acoustic features extracted from the live recorded audio. The system recognizes emotions using 13 Mel Frequency Cepstral Coefficients which are extracted from the speech signal. The extracted features are fed in to 4 layer two-dimensional convolutional neural network (2D CNN) which has been already deployed in android application to recognize the emotion of the recorded audio.

Model is tuned to recognize five classes of emotions namely Happy, Sad, Fear, Angry and Calm with 85% accuracy. Our system is language independent in recognising the emotions. The accuracy of results obtained from testing various movie samples and conversational speech from different languages shows that the system has achieved more than the state-of-art results and may gain significance in the range of applications.

ABSTRACT

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

CNN	Convolutional Neural Network
MFCC	Mel Frequency Cepstral Coefficients
SER	Speech Emotion Recognition
HCI	Human Computer Interactions
GMM	Gaussian Mixture Model
UI	User Interface

CHAPTER 1

INTRODUCTION

1.1 PROBLEM DOMAIN

Speech emotion recognition (SER) is the natural and fastest way of exchanging and communication between humans and computers and plays an important role in real-time applications of human-machine interaction. The speech signals generated for SER is an active area of research in digital signal processing. It used to recognize the qualitative emotional state of speakers using speech signals, which has more information than spoken words. Emotion plays a significant role in daily interpersonal human interactions. This is essential to our rational as well as intelligent decisions. Speaker and culture dependency is a very significant aspect in this system. A certain emotion expressed is highly dependent on the speaker and his or her age, language, culture, personality, environment, etc. The speech emotion recognition involves analysis of the speech signal to identify the appropriate emotion based on training its features such as pitch, formant and phoneme. There may be more than one perceived emotion in the same utterance and it is very important to determine this emotional state. It is drawing more attention in the applications where emotion recognition eases the speaker identification and mental status, such as in criminal investigation, intelligent assistance, detecting frustration, disappointment, surprise/amusement, health care and medicine and a better Human Computer Interface.

In this project, we used an android application with deep neural network model to recognize the emotion from speech signal irrespective of languages and gender.

1.2 PROBLEM DESCRIPTION

On recording a human voice or movie sample using the android application " Finding emo", the system should extract the spectral features of speech signal and recognize the emotion which falls under any of the five emotion classes - Happy, sad, fear, angry and calm using the trained deep neural network model.

1.3 SCOPE

The system aims to improve HCI by taking into consideration the wide range of subtle human emotions and classifying them efficiently aiming for a greater accuracy. The proposed system extends to classify - Happy, sad, fear, angry and calm emotions.

1.4 CHALLENGES IN RECOMMENDED SYSTEM

Since there are a multitude of human emotions, one of the key challenges was to distinguish effectively between the overlapping emotions with almost similar tones. Some cases of overlapping emotions are neutral and sad (low tones), happy and angry (high pitched tone). The system is designed to use energy level features to accurately classify even the overlapping samples to reach an even better accuracy of 83%.

1.5 KEY CONTRIBUTIONS

The human emotion recognition process collects data from audio datasets in which audio files are to be in .wav format. The signal is then split into short frames and preprocessed to classify voiced and unvoiced signals. The Spectral features like MFCC (Mel Frequency Cepstral Coefficient) yielding 13 energy specific coefficients are extracted. A 4 layer two-dimensional Convolutional neural network (2D CNN) is constructed to learn emotion-related features from speech using the extracted features. The classification of trained data is based on the emotion classes such as happy, sad, anger, fear and neutral. Then the system is tested using data samples from different language languages like English, Tamil, Hindi and Malayalam. Since the system works based on the energy specific features of an audio and is independent of any language, it yielded a fairly high accuracy.

1.6 ORGANISATION OF THESIS

Chapter 2 discusses the existing approaches for speech emotion recognition particularly using acoustic features of speech. It also analyses the advantages and disadvantages of each approach. Chapter 3 explains the overall system architecture and the design of various modules along with their complexity. Chapter 4 gives the implementation details of each module, describing the algorithms used. Chapter 5 elaborates on the results of the implemented system and gives an idea of its efficiency. It also contains information about the dataset used for testing and other observations made during testing. Chapter 6 concludes the thesis and gives an overview of its criticisms. It also states the various extensions that can be made to the system to make it function more effectively.

CHAPTER 2

RELATED WORK

2.1 DEEP LEARNING TECHNIQUES

Deep Learning techniques have been recently proposed as an alternative to traditional techniques in SER. R. A. Khalil, et al., [5] has provided a detailed review of the deep learning techniques for SER. Deep learning techniques such as DBM, RNN, DBN, CNN, and AE and their layer-wise architectures are briefly elaborated based on the classification of various natural emotion such as happiness, joy, sadness, neutral, surprise, boredom, disgust, fear, and anger. It has been said that these methods offer easy model training as well as the efficiency of shared weights. Deep convolution neural network produced better results than other traditional SER techniques. Some Acoustic Variations Observed Based on Emotions is also summarised. As it has been proved that Deep CNN (DCNN) produced the best results in predicting the emotion, 4 layer 2D Convolution neural network is adopted in our system to train the model.

Mustaqeem and Kwon, et al., [1] used a CNN architecture with some salient features extraction mechanism to improve the accuracy and achieve reduced computational complexity of the overall SER model.A dynamic adaptive threshold technique is used to remove noise and silent signals from speech signals. Then the enhanced speech signals are converted to spectrograms to increase the accuracy and decrease the computational complexity of the proposed model. A stride CNN architectures for SER using spectrograms to learn most salient and discriminative features in a convolutional layer using some special stride set to down-sample feature maps rather than pooling layers is used. An artificial intelligence-assisted deep stride convolutional neural network (DSCNN) architecture using the plain nets strategy to learn salient and discriminative features from spectrogram of speech signals that are enhanced in prior steps to perform better. Local hidden patterns are learned in convolutional layers with special strides to down-sample the feature maps rather than pooling layer and global discriminative features are learned in fully connected layers. A SoftMax classifier is used for the classification of emotions in speech. Similarly, a 4 layer 2D CNN architecture with different noise removal technique is used in our system as model summary is shown in figure 4.9.

Y. Xie, et al., [13] proposed a novel method for speech recognition using frame-level speech features combined with attention-based long short-term memory (LSTM) recurrent neural networks to make full use of the difference of emotional saturation between time frames. Frame-level speech features were extracted from waveform to replace traditional statistical features, which could preserve the timing relations in the original speech through the sequence of frames. Proposed algorithm reduces the computational complexity by modifying the forgetting gate of traditional LSTM and in the final output of the LSTM, an attention mechanism is applied to both the time and the feature dimension to obtain the information related to the task, rather than using the output from the last iteration of the traditional algorithm. The experiments demonstrate that the new attention-gate can also improve the recognition rate. An improved attention-based LSTM is proposed for emotion classification.

2.2 FEATURE EXTRACTION

Ramdinmawii, et al., [12] analyzed the four basic emotions (Anger, Happy, Fear and Neutral) from emotional speech signals. Signal processing methods are used for obtaining the production features from these signals. The analysis of these emotion states has been done using features, namely, instantaneous fundamental frequency (F0) using Zero Frequency Filtering, Formant frequencies (F1, F2, F3) using LP spectrum, signal energy, and dominant frequencies. Short-time signal energy (STE) and ZCR are obtained in the voiced and unvoiced regions using a rectangular window of 200 samples.

Lukose, et al., [8] proposed a music system which recognizes five emotions- anger, anxiety, boredom, happiness and sadness. This system uses the STE and ZCR for separation of voiced and unvoiced signal and extracts MFCC features for emotion recognition. Once the emotion of the speech is recognized, the system platform automatically selects a piece of music as a cheer up strategy from the database of song playlist stored. The analysis results show that this SER system implemented over five emotions provides successful emotional classification performance of 76.31% using GMM model and an overall better accuracy of 81.57% with SVM model.

M. S. Likitha, et al., [6] also used Mel Frequency Cepstral Coefficient (MFCC) technique to recognize emotion of a speaker from their voice. The designed system was validated for Happy, sad and anger emotions and the efficiency was found to be about 80%.

Ram, et al., [11] proposed a system using two features for emotion recognition as linear predictive coding (LPC) and spectral analysis. As a result, the speech waveforms are commonly split into small frames (typically 5 ms to 40 ms) in which the signal characteristics are considered quasi-stationary to allow for short-term spectral analysis and feature extraction. This parametric representation of speech is used to generate input feature to the recognition models. This system recognizes human emotion by means of spectral analysis in which human emotion speech can be determined due to their different spectral peak.

Patel, et al., [9] sought to describe emotional expressions according to physiological variations measured from the inverse-filtered glottal waveform in addition to standard parameter extraction. An acoustic analysis was performed on a subset of the /a/ vowels. Subsequent principal components analysis revealed the three components that explain acoustic variations due to emotion, including “tension” (CQ, H1-H2, MFDR, LTAS) “perturbation” (jitter, shimmer, HNR), and “voicing” (fundamental frequency).

Parselmouth [4], an open-source Python library that facilitates access to core functionality of Praat - a system for doing phonetics by computer [2] in Python, in an efficient and programmer-friendly way. Parsel mouth is used in extracting features from audio signal.

Having a generic approach to SER, we can identify the emotion. It has been observed that noise removal and voiced region separation using ZCR and STE produced good results. Commonly, MFCC features improves the system in predicting the emotion. Thus from the analysis of related works in predicting the emotion, our system is designed to use MFCC features and 2D CNN to recognize the emotion. Additionally, our system is deployed in real time (Android application) to gain significance in real time. Our system handles any languages for recognition. Thus there is a lot of scope for our system.

CHAPTER 3

SYSTEM DESIGN

3.1 SYSTEM ARCHITECTURE

The system aims at storing the audio signal in device storage and recognizing the emotion from speech signal, given a recorded speech as input. The block diagram of the overall system is shown in figure 3.1. The emotion recognition system starts its processing by recording audio in raw PCM format. It is then converted into .wav format by adding proper header to audio files. The signal is then split into short frames and then preprocessed for noise removal using Webrtcvad and then voiced and unvoiced signals are separated. Spectral Features using Mel Frequency Cepstral Coefficient with 13 energy specific coefficients is extracted librosa python library. The extracted features are fed in to deployed deep neural network constructed using a 4 layer two-dimensional convolutional neural network (2D CNN) model using keras and tensorflow. The Neural network model is constructed to learn emotion-related features from speech with the highest possible accuracy. Chaquopy is used to integrate python with android and bundle trained .h5 model with apk. Finally the predicted emotion falls under five distinct emotion classes (Angry, Fear, Happy, Sad and Neutral).

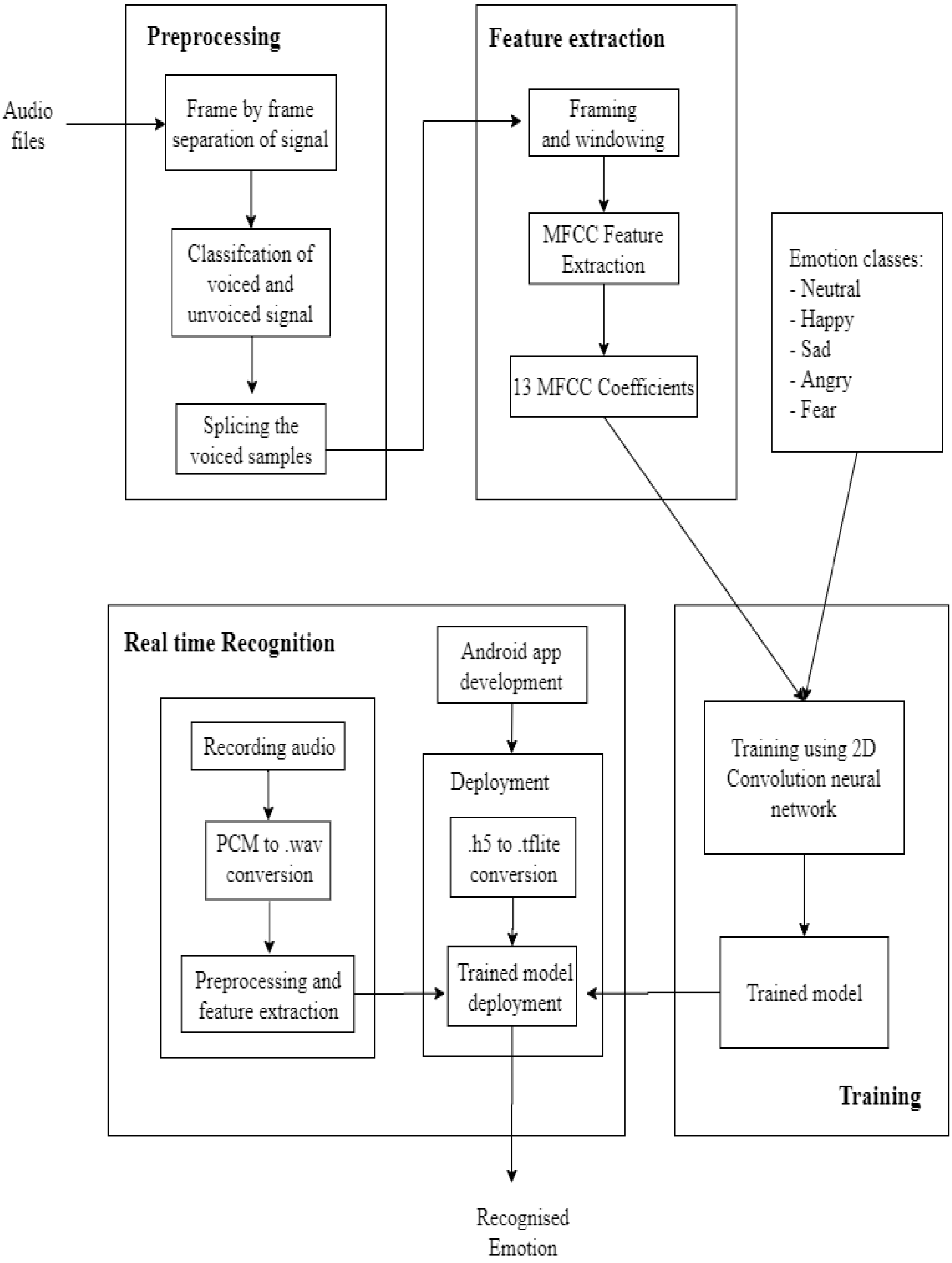


Figure 3.1: System Architecture

3.2 MODULE DESIGN

3.2.1 Dataset details

3.2.1.1 RAVDESS

The Ryerson Audio-Visual Database of Emotional Speech and Song [7]. Each of the 1440 RAVDESS speech only files has a unique filename. 60 trials per actor x 24 actors= 1440. The filename consists of a 7-part numerical identifier (e.g.,02-01-05-01-02-01-12.wav). These identifiers define the stimulus characteristics like Modality (full-AV, video-only, audio-only), channel (speech, song), Emotion (Neutral, Calm, Happy, Sad, Angry, Fearful, Disgust, Surprised), Emotional intensity, Statement, Repetition and Actor (male, female).

3.2.1.2 SAVEE

Surrey Audio-Visual Expressed Emotion [3]. Database has emotions described psychologically in discrete categories: anger, disgust, fear, happiness, sadness and surprise. This consists of 15 sentences per emotion: 3 common, 2 emotion-specific, 10 generic sentences for each emotion along with 30 neutral sentences giving a total of 480 sentences.

3.2.1.3 TESS

Toronto emotional speech set [10]. A set of 200 target words were spoken in the carrier phrase ”Say the word _____ by two actresses (aged 26 and 64 years) and recordings were made of the set portraying each of seven emotions (anger, disgust, fear, happiness, pleasant, surprise, sadness, and neutral). There are 2800 stimuli in total.

3.2.2 Preprocessing

The input to this module is the audio sample in .wav format. The audio file is loaded in terms of floating point time series. Webrtcvad, a python wrapper is used to separate the voiced signal using multiple frequency band features with Gaussian Mixture Model (GMM).Voiced signals are spliced together by webrtcvad. The detailed module design is shown in figure 3.2.

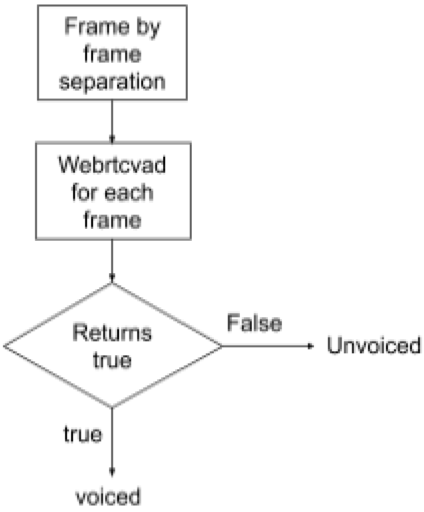


Figure 3.2: Preprocessing

3.2.3 Feature extraction

This module is to extract the 13 Mel frequency cepstral coefficients (MFCC). Pre-emphasis - a filtering procedure which maximizes signal energy is applied on voiced signals. Those signals undergo frame blocking where the continuous speech signal is divided into frames of N samples with adjacent frames being separated by M (N $>$ M). Figure 3.3 shows the frame blocking process.

Pseudocode for frame blocking is as follows:

```
Framing(Sampled_audio)
    Frame_length=40ms
    Frame_shift= 10ms
    i=0
    Frames[]
    While i+Frame_length < total_samples:
        Frame= Sampled_audio[i to i+Frame_length]
        i=i+Frame_shift
        Add Frame to Frames
    Return Frames
```

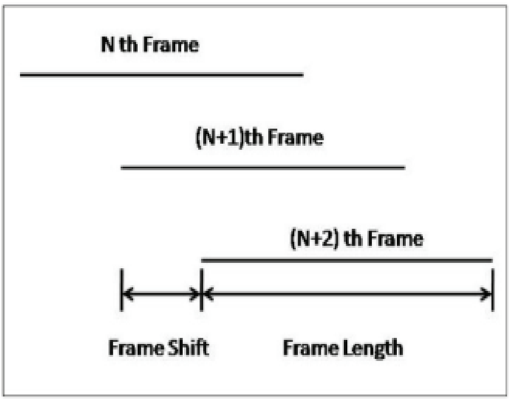


Figure 3.3: Frame blocking

Frame blocking is followed by windowing where each individual frame is windowed to minimize the signal discontinuities at the beginning and end of each frame using a hamming window. Windowing means multiplying the window function w(n) with the framed speech signal s(n) to obtain the windowed speech signal sw(n). Figure 3.4 shows the windowing process.

Window_Function = Hamming Window (Window_length = Frame_length)

Windowed_Frames = Frames x Window_Function

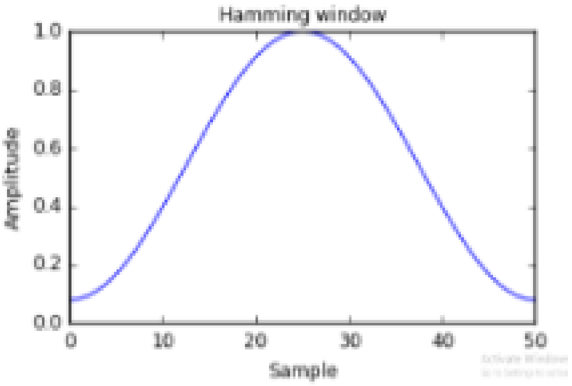


Figure 3.4: Windowing

MFCC features are extracted from the windowed signals. The spectral information is converted to MFCC by passing the signals through band pass filters where higher frequencies are artificially boosted, and then applying an inverse Fast Fourier Transform (FFT) on it. Figure 3.5 shows the feature extraction process. Table 3.1 shows the 13 energy coefficients of MFCC.

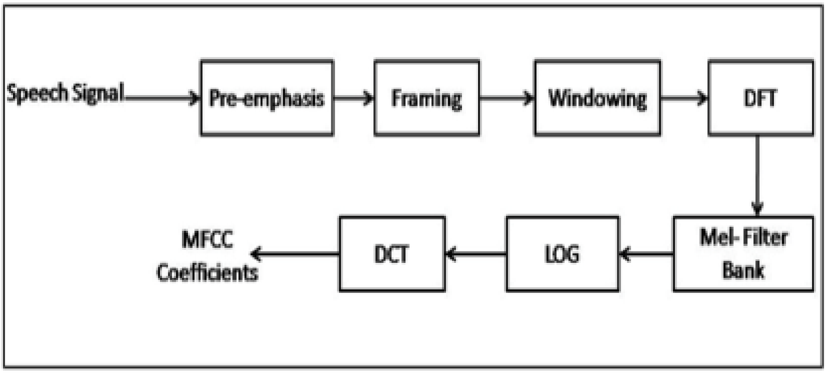


Figure 3.5: MFCC extraction

S.no	Name of the feature
1	Energy
2	Energy of first level decomposed band
3	Energy intensity
4	Energy entropy
5	Energy for sub band
6	Jitter
7	Timbre
8	Duration
9	Average duration of every successive energy
10	Energy duration (original and decomposed band)
11	Gain of sampling frequency
12	Energy first band
13	Energy second band

Table 3.1: MFCC features

3.2.4 Training phase

The input to the model is feature vector with dimension (13 , 16). Input vector with 13 MFCC coefficients is fed into the neural network as training parameters. Input vector is passed to the 4 - 2D Convolution layers with filter count (16, 32, 64, 128), a 3 x 3 kernel , stride 1 and relu activation function. Max Pooling (2 x 2 pool with Stride 2) for dimensionality reduction. Flattening 3D to 1D vectors with 0.5 dropout_rate to ensure independence avoiding overfitting. Three fully connected layers with relu and softmax activation finally yielding the output probability The output will be the trained model with five emotion classes - Happy, sad, fear, angry and neutral. Figure 3.6 shows the neural network architecture used in this system to train the model.

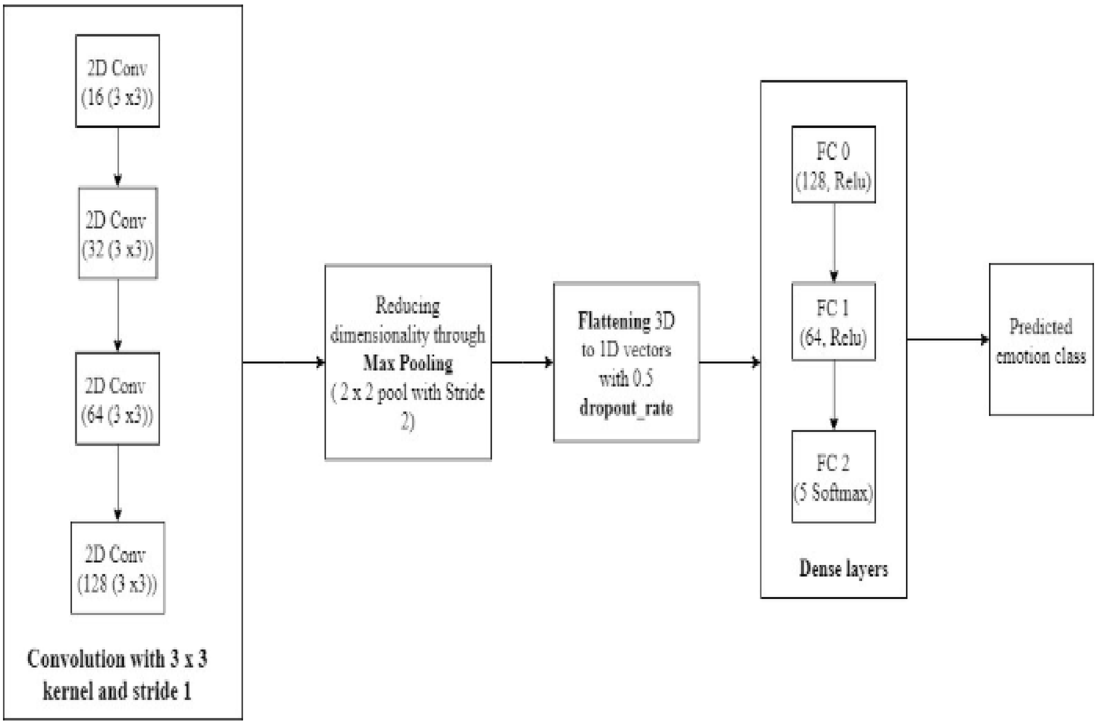

2D Conv (16 (3 x3))
2D Conv (32 (3 x3))
2D Conv (64 (3 x3))
2D Conv (128 (3 x3))
Convolution with 3 x 3 kernel and stride 1
Reducing dimensionality through Max Pooling (2 x 2 pool with Stride 2)
Flattening 3D to 1D vectors with 0.5 dropout_rate
FC 0 (128, Relu)
FC 1 (64, Relu)
FC 2 (5 Softmax)
Dense layers
Predicted emotion class

Figure 3.6: Neural network architecture

3.2.5 Android application

This module is to record the audio through android application, preprocessing and extracting the features from audio and predict the emotion. This module has the phases include UI designing, recording audio, conversion from PCM to WAV format, preprocessing, feature extraction, deployment of trained model in android application and Emotion recognition.

Trained model is deployed into the android application using chaquopy - a tool used to create python instance to be called from java. Pre trained keras .h5 model is bundled with python script which is invoked by creating a python instance in java file.

Android has a built-in microphone through which can capture audio and store it , or play it on the phone. Audio is recorded with the help of android application ” Finding emo” in PCM format (raw file). PCM file is converted into .wav file by writing the header and filling it with raw PCM bytes from AudioRecord until it reaches a certain size or is stopped by the user. Then the WAV header is updated to include the proper final chunk sizes and writes the proper 44-byte RIFF/WAVE header to/for the given stream followed by updating the given wav file’s header to include the final chunk size. The output of this submodeule is .wav audio file. .wav file is preprocessed for noise removal and separation of voiced and unvoiced regions. 13 Mel frequency cepstral coefficients are extracted using librosa python library. The extracted features are fed into the deployed trained model, inference on input data is performed and recognised emotion is displayed in application. Meanwhile, the recorded audio overwrites the previously recorded audio in the device storage.

CHAPTER 4

SYSTEM DEVELOPMENT

4.1 IMPLEMENTATION

4.1.1 Preprocessing and feature extraction

The dataset for training of the model in the system consists of manually collected audio files and from 3 datasets namely RAVDESS, SAVEE and TESS. The sample audio file which is going to be processed is shown in the figure 4.1.

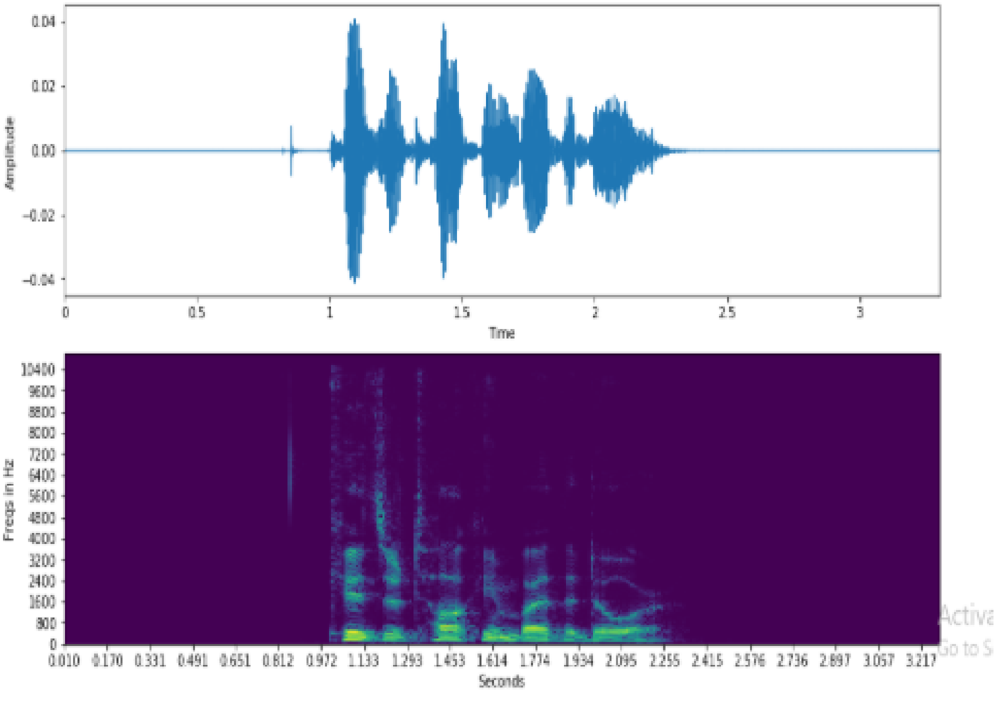


Figure 4.1: Sample audio file

The collected audio files are Pre-processed to train the model. Removal of noise and separation of voiced and unvoiced region is done using the webrtcvad package.This is a python interface to the WebRTC Voice Activity Detector (VAD). A VAD classifies a piece of audio data as being voiced or unvoiced. The Pre-processed audio file is shown in the figure 4.2. The code implementation for preprocessing the audio file is as follows:

Pre processing

```
vad = webrtcvad.Vad(int(args[0]))
segments = vad_collector(sample_rate, 30, 300, vad, frames)
for i, segment in enumerate(segments):
  write_wave(path, segment, sample_rate)
```

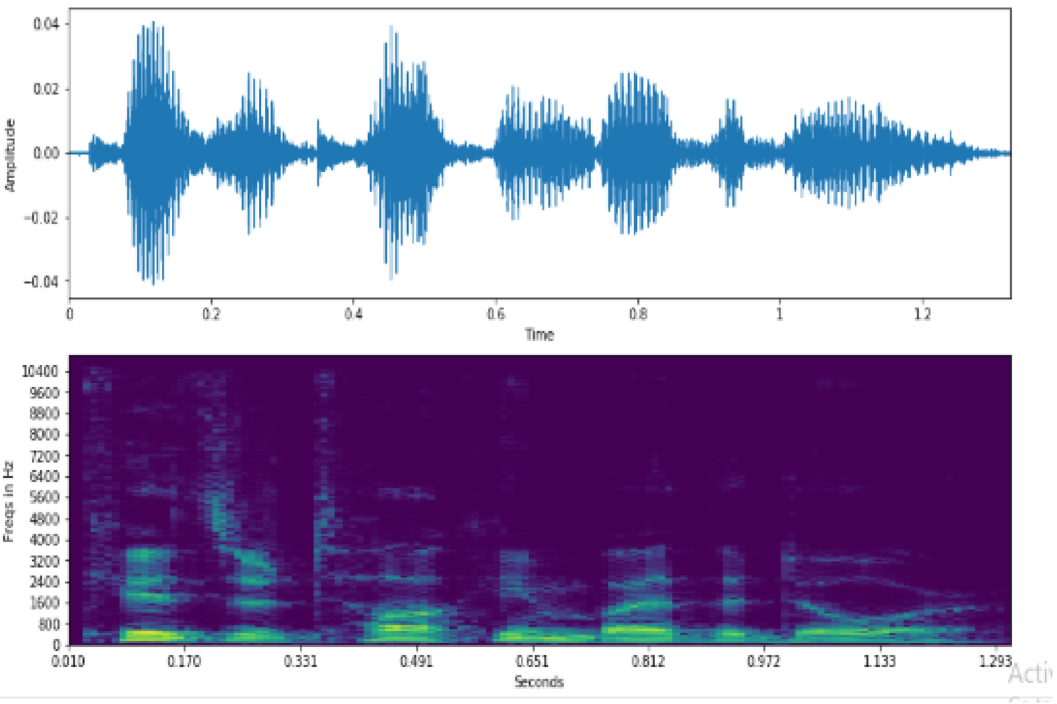


Figure 4.2: Pre-processed audio file

The 13 MFCC coefficients are extracted from the collected voiced signal for further processing. LibROSA which is a python package for music and audio analysis is used for feature extraction. The extracted MFCC coefficients are shown in the figure 4.4. Python script to extract the MFCC features from voiced signal is as follows:

Feature extraction

```
1. S=librosa.feature.melspectrogram(aa,sr=sample_rate,
   n_mels=128)
2. log_S = librosa.power_to_db(S, ref=np.max)
3. mfcc = librosa.feature.mfcc(S=log_S, n_mfcc=13)
4. delta2_mfcc = librosa.feature.delta(mfcc, order=2)
```

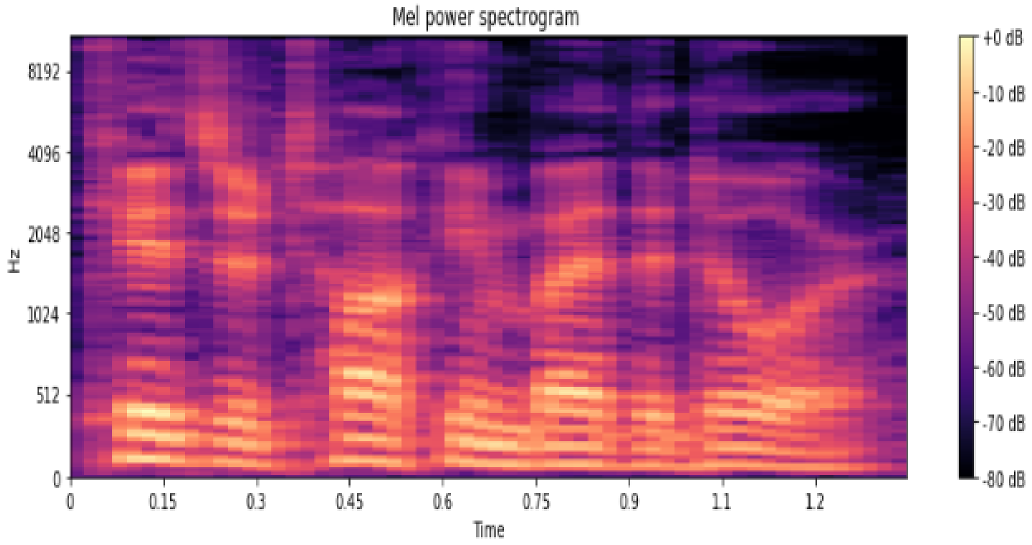


Figure 4.3: Mel power spectrogram

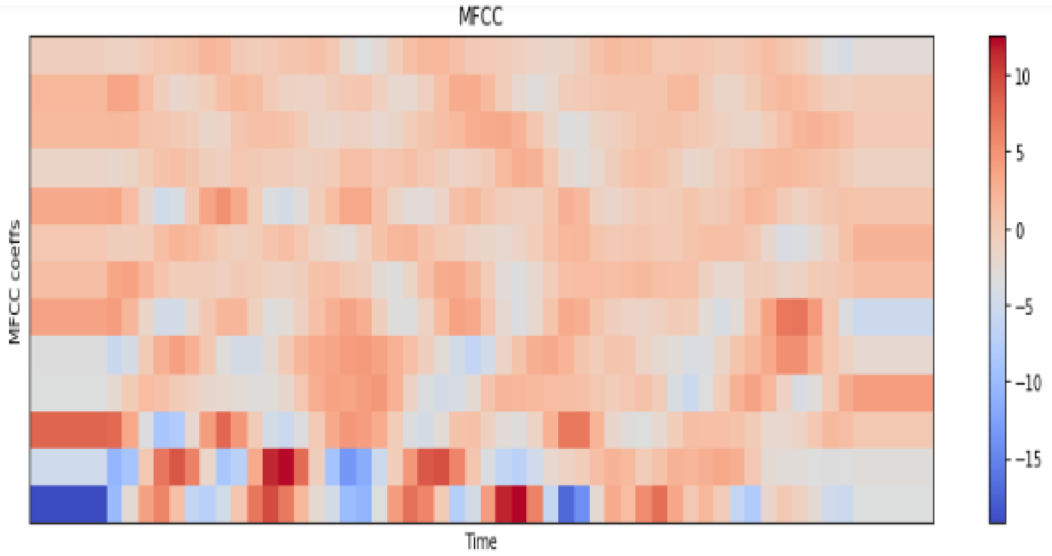
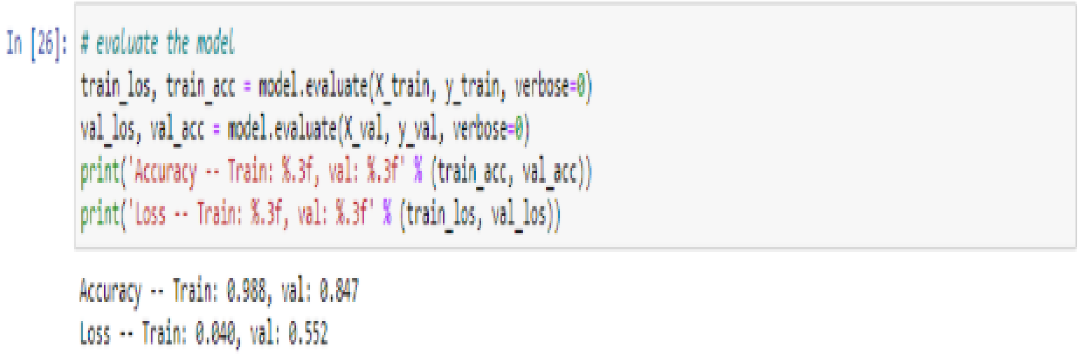


Figure 4.4: Extracted MFCC features

4.1.2 Training CNN model

The extracted coefficients are fed into 4 layer 2D CNN network for training the audio samples for emotion recognition. Keras and tensorflow is used for training the model. Model summary is shown in the figure 4.9. Our trained model achieves the 98% accuracy with 0.04% loss as shown in the figure 4.5. Loss and accuracy in duration of 5 epochs is shown in figures 4.6, 4.7 and 4.8



```
In [26]: # evaluate the model
         train_los, train_acc = model.evaluate(X_train, y_train, verbose=0)
         val_los, val_acc = model.evaluate(X_val, y_val, verbose=0)
         print('Accuracy -- Train: %.3f, val: %.3f' % (train_acc, val_acc))
         print('Loss -- Train: %.3f, val: %.3f' % (train_los, val_los))
```

```
Accuracy -- Train: 0.988, val: 0.847
Loss -- Train: 0.040, val: 0.552
```

Figure 4.5: Trained model

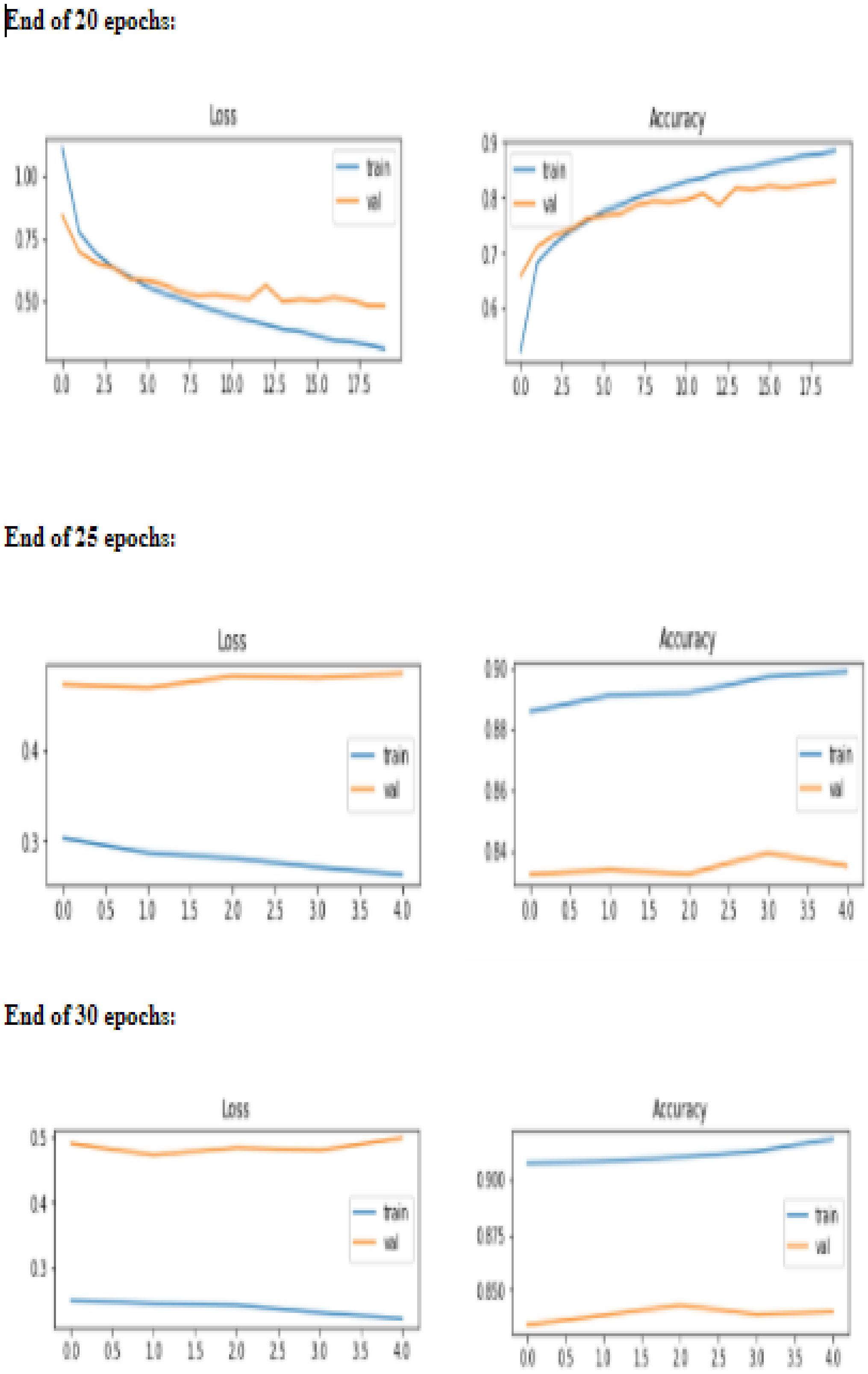
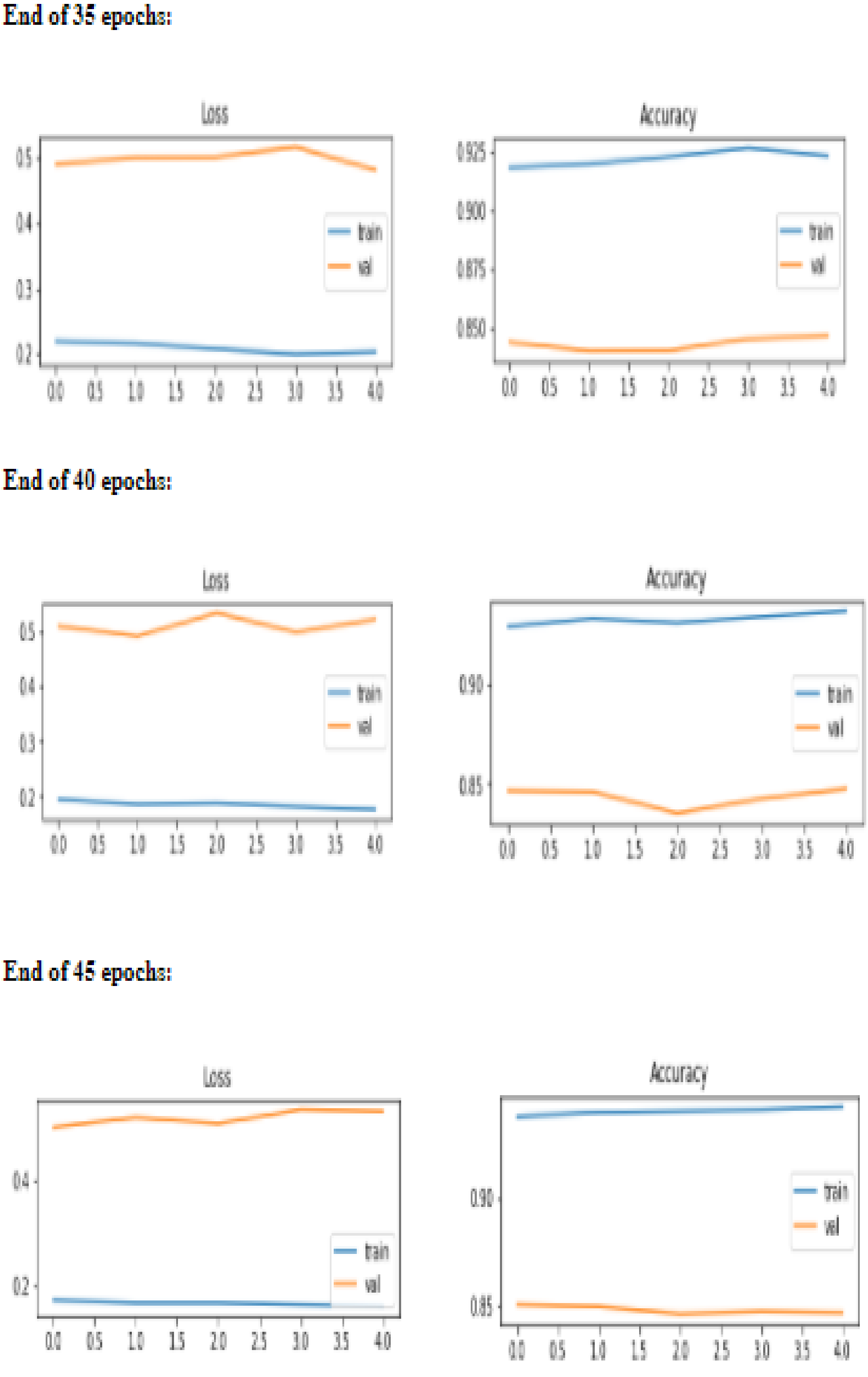


Figure 4.6: Epoch 20, 25, 30 details

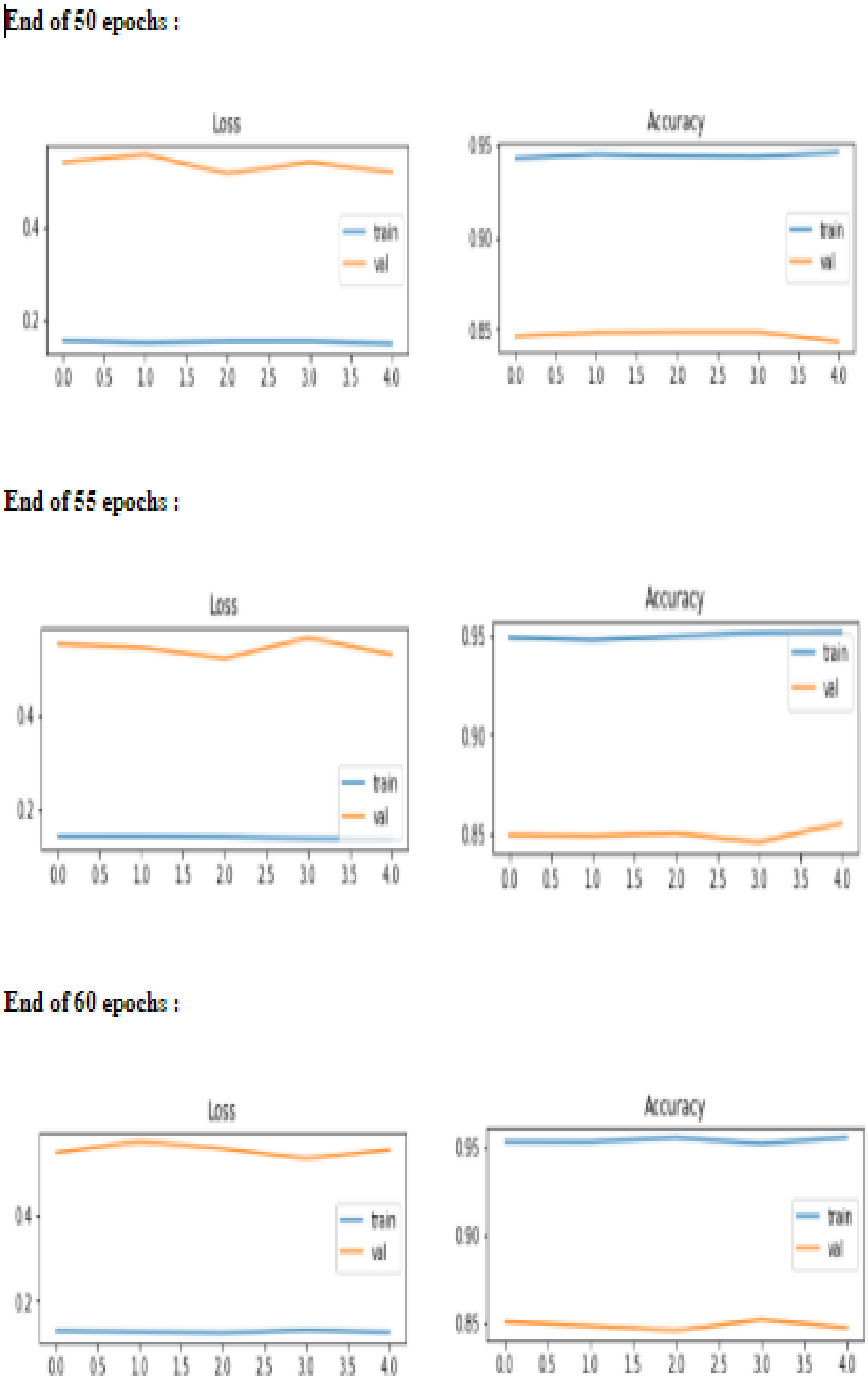


End of 35 epochs:

End of 40 epochs:

End of 45 epochs:

Figure 4.7: Epoch 35, 40, 45 details

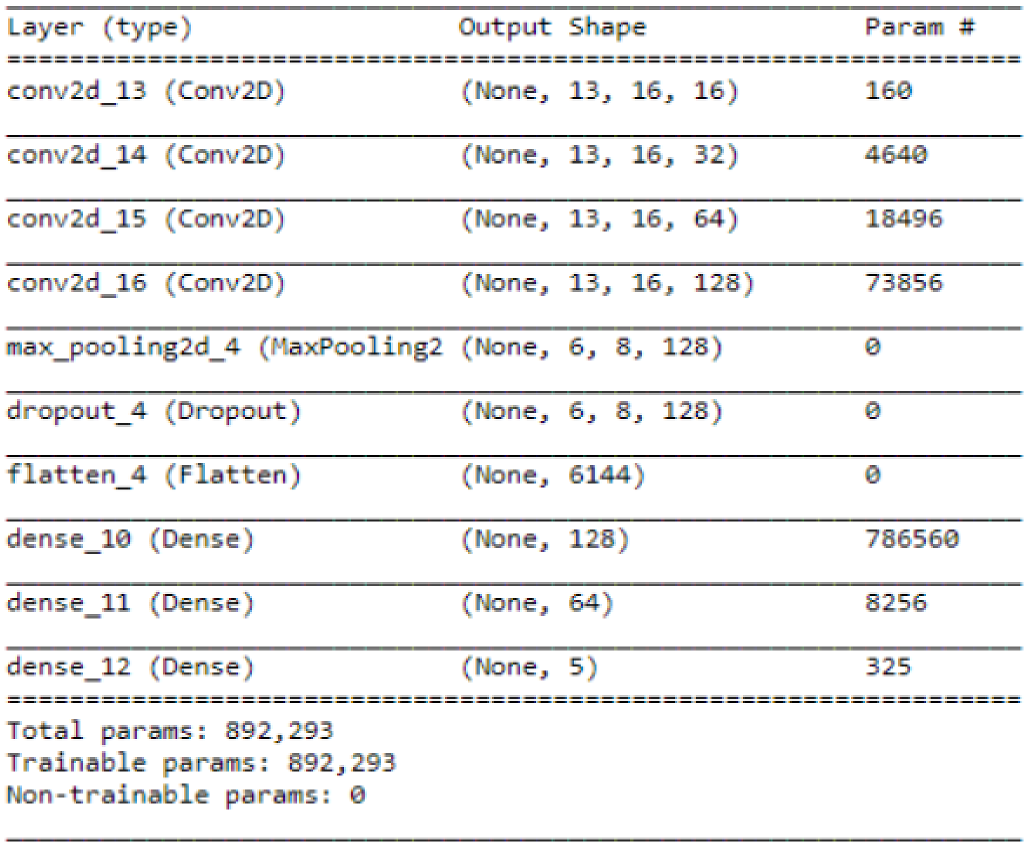


End of 50 epochs :

End of 55 epochs :

End of 60 epochs :

Figure 4.8: Epoch 50, 55, 60 details



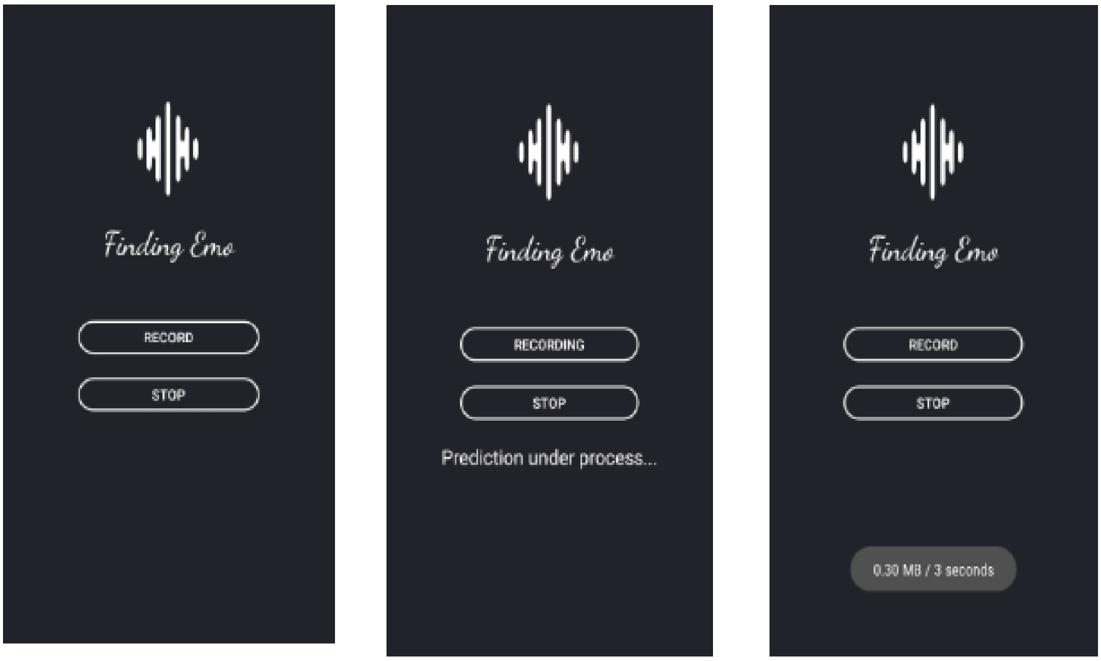
```
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
conv2d_13 (Conv2D)           (None, 13, 16, 16)        160       
_________________________________________________________________
conv2d_14 (Conv2D)           (None, 13, 16, 32)        4640      
_________________________________________________________________
conv2d_15 (Conv2D)           (None, 13, 16, 64)        18496     
_________________________________________________________________
conv2d_16 (Conv2D)           (None, 13, 16, 128)       73856     
_________________________________________________________________
max_pooling2d_4 (MaxPooling2 (None, 6, 8, 128)         0         
_________________________________________________________________
dropout_4 (Dropout)          (None, 6, 8, 128)         0         
_________________________________________________________________
flatten_4 (Flatten)          (None, 6144)              0         
_________________________________________________________________
dense_10 (Dense)             (None, 128)               786560    
_________________________________________________________________
dense_11 (Dense)             (None, 64)                8256      
_________________________________________________________________
dense_12 (Dense)             (None, 5)                 325       
=================================================================
Total params: 892,293
Trainable params: 892,293
Non-trainable params: 0
_________________________________________________________________
```

Figure 4.9: Model summary

4.1.3 Deployment in realtime

This section shows the real time implementation. The deployment of the system requires python packages such as librosa, keras, tensorflow, webrtcvad etc. Android IDE like Android studio can be used to deploy the system successfully. Chaquopy - A tool used to integrate python with java and invoke python script using python instance created from java file.

The recorded audio file was given to the system as input. The UI indicates that the audio is getting recorded and stored in device storage as shown in figure 4.10b and 4.10c respectively. Figure 4.10a shows the launch activity of android application ”Finding emo”.

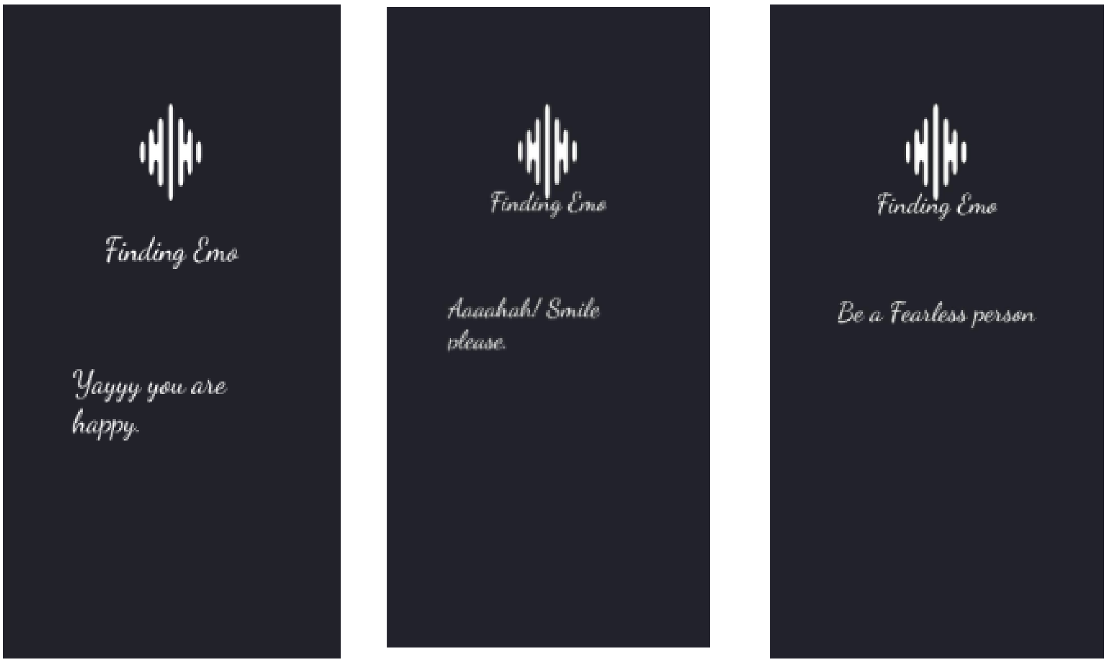


Prediction under process...

0.30 MB / 3 seconds

(a) Launch activity (b) Recording audio (c) Stop audio

Figure 4.10: UI Android application - Finding emo



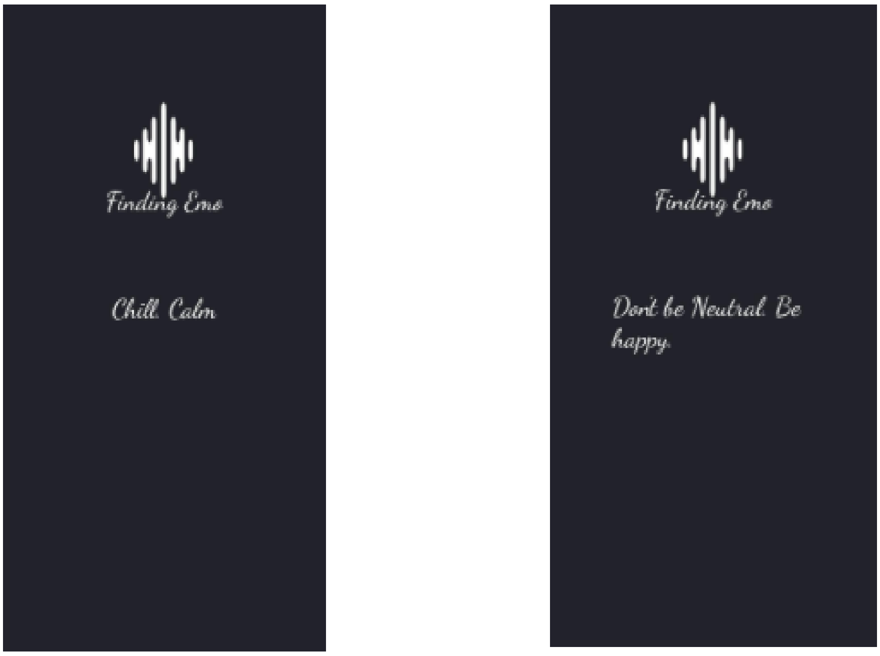
Yayyy you are happy.

Aaaahah! Smile please.

Be a Fearless person

(a) Happy (b) Sad (c) Fear

Figure 4.11: UI for Emotions Happy, Sad and Fear



Finding Emo

Chill. Calm

Finding Emo

Don't be Neutral. Be happy.

(a) Angry

(b) Neutral

Figure 4.12: UI for Emotions Angry and Neutral

An User interface of app for each emotion prediction, Happy , sad, fear, angry and calm is shown in figures 4.11a, 4.11b, 4.11c, 4.12a and 4.12b respectively. The output screen shows the recognized emotion. The system is tested for various test cases which are detailed below.

4.2 TEST CASES FOR EACH MODULE

This section provides the test cases for each of the modules of the system developed. Table 4.1 shows the number of samples (535 in total) in each language and in each emotions used during testing. Emotion prediction is done using the probability value that is predicted for each emotion. Emotion with higher prediction value is recognised as final emotion.

Language	No of samples					
	Neutral	Happy	Sad	Angry	Fear	Total
English	6	57	13	23	21	120
Tamil	26	19	27	57	20	149
Hindi	23	34	30	56	25	168
Malayalam	26	17	16	29	10	98

Table 4.1: Number of samples in each language

4.2.1 Preprocessing

4.2.1.1 Test Pre-requisite

A recorded audio file in .wav format as input audio file.

4.2.1.2 Description

The set of test cases to this module covers audio files in any languages, time duration $>$ 1 second and in any gender baseline.

4.2.1.3 Test Cases

- **TC_ID:** 01

  Input: Audio file in .wav format.

  Expected Output: Voiced audio signal with proper .wav header
- **TC_ID:** 02

  Input: Audio in other formats like raw audio (PCM).

  Expected Output: Attaches .wav header files to PCM audio chunks and produces the voiced audio sample.
- **TC_ID:** 03

  Input: Audio with no human voice detected.

  Expected Output: Produces the maximum possible voiced region.

4.2.2 Feature Extraction

4.2.2.1 Test Pre-requisite

Preprocessed audio sample as input audio file.

4.2.2.2 Description

MFCC coefficients are extracted from voiced region of recorded audio sample.

4.2.2.3 Test Cases

- **TC_ID:** 01
  Input: Audio signal with any duration of voiced region.
  Expected Output: 13 MFCC coefficients.

4.2.3 Training the model

4.2.3.1 Test Pre-requisite

13 MFCC coefficients spectogram with target emotion.

4.2.3.2 Description

The set of test cases to this module covers different durations of audio in different languages.

4.2.3.3 Test Cases

- **TC_ID:** 01
  Input: Audio samples in various languages.
  Expected Output: Trained model.

4.2.4 Real time recognition

4.2.4.1 Test Pre-requisite

Android app with deployed trained model and recorded audio.

4.2.4.2 Description

The set of test cases covers all forms of human speech.

4.2.4.3 Test Cases

- **TC_ID:** 01

  Input: Recorded audio.

  Expected Output: Predicted emotion.
- **TC_ID:** 02

  Input: Rerecording audio without stopping previous audio recording.

  Expected Output: Throws error - task already running.
- **TC_ID:** 03

  Input: Angry sample

  Intermediate output: Probability array

Neutral	Happy	Sad	Angry	Fear
8.67E-08	1.62E-06	1.71E-05	0.9210	0.0788

  Expected Output: Chill. Calm
- **TC_ID:** 04

  Input: Happy sample

  Intermediate output: Probability array

Neutral	Happy	Sad	Angry	Fear
3.82E-18	0.9999	5.39E-18	2.19E-05	1.05E-14

  Expected Output: Yayyy! You are happy.

- **TC_ID:** 05

  Input: Sad sample

  Intermediate output: Probability array

Neutral	Happy	Sad	Angry	Fear
0.1721	0.0144	0.7533	0.0002	0.0598

  Expected Output: Aahahh! Smile please.

- **TC_ID:** 06

  Input: Fear sample

  Intermediate output: Probability array

Neutral	Happy	Sad	Angry	Fear
6.43E-07	0.0008	0.0001	0.0015	0.9974

  Expected Output: Be a fearless person.

- **TC_ID:** 07

  Input: Neutral sample

  Intermediate output: Probability array

Neutral	Happy	Sad	Angry	Fear
0.5989	0.0175	0.3689	0.0038	0.0106

  Expected Output: Don’t be neutral. Be happy.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 RESULTS OBTAINED DURING TESTING

Following tables 5.2, 5.3, 5.4 and 5.5 shows some of the audio samples used for testing. Conventions used : 0 - Neutral, 1 - Happy, 2 - Sad, 3 - Angry, 4 - Fear. It has been observed from the results that overlapping occurs between similar energy audio samples such as Happy - Anger, Sad - neutral which led to misprediction of emotion. Table 5.1 shows some of the ambiguous samples.

Sample name	Predicted		Description
	Value	Emotion	
aslan.wav	3	angry	Although a war greeting of long life, the tone in this sample is angry and hence the system predicts it so.
choru_2.wav	3	angry	This sad cry sample has high energy tone as that of an angry emotion
movie1.wav	2	sad	An advice type neutral sentence said in a sad tone
ladybug.wav	3	angry	a quick surprise tone with the sample ending with a shouting cry
main_tujhe_1.wav	4	fear	A challenging sentence said in a way of inflicting fear

Table 5.1: Ambiguous samples

Sample name	Actual		Predicted	
	Value	Emotion	Value	Emotion
English_1.wav	3	angry	1	happy
English_6.wav	3	angry	3	angry
English_9.wav	3	angry	3	angry
English_12.wav	3	angry	3	angry
English_40.wav	3	angry	4	fear
English_62.wav	4	fear	3	angry
English_79.wav	4	fear	0	neutral
English_87.wav	4	fear	4	fear
English_92.wav	4	fear	4	fear
English_99.wav	4	fear	4	fear
English_3.wav	1	happy	1	happy
English_4.wav	1	happy	4	fear
English_10.wav	1	happy	1	happy
English_11.wav	1	happy	1	happy
English_35.wav	1	happy	3	angry
English_2.wav	0	neutral	1	happy
English_5.wav	0	neutral	0	neutral
English_43.wav	0	neutral	0	neutral
English_95.wav	0	neutral	0	neutral
English_96.wav	0	neutral	0	neutral
English_7.wav	2	sad	1	happy
English_8.wav	2	sad	2	sad
English_23.wav	2	sad	2	sad
English_48.wav	2	sad	3	angry
English_55.wav	2	sad	2	sad

Table 5.2: Tested samples - English

Sample name	Actual		Predicted	
	Value	Emotion	Value	Emotion
Tamil_9.wav	3	angry	3	angry
Tamil_12.wav	3	angry	4	fear
Tamil_13.wav	3	angry	3	angry
Tamil_15.wav	3	angry	1	happy
Tamil_17.wav	3	angry	4	fear
Tamil_3.wav	4	fear	4	fear
Tamil_4.wav	4	fear	3	angry
Tamil_6.wav	4	fear	4	fear
Tamil_10.wav	4	fear	4	fear
Tamil_14.wav	4	fear	4	fear
Tamil_19.wav	1	happy	4	fear
Tamil_29.wav	1	happy	1	happy
Tamil_30.wav	1	happy	3	angry
Tamil_32.wav	1	happy	1	happy
Tamil_59.wav	1	happy	2	sad
Tamil_106.wav	0	neutral	2	sad
Tamil_107.wav	0	neutral	2	sad
Tamil_108.wav	0	neutral	0	neutral
Tamil_109.wav	0	neutral	1	happy
Tamil_110.wav	0	neutral	0	neutral
Tamil_11.wav	2	sad	0	neutral
Tamil_20.wav	2	sad	2	sad
Tamil_21.wav	2	sad	4	fear
Tamil_22.wav	2	sad	2	sad
Tamil_23.wav	2	sad	2	sad

Table 5.3: Tested samples - Tamil

Sample name	Actual		Predicted	
	Value	Emotion	Value	Emotion
Hindi_17.wav	3	angry	1	happy
Hindi_19.wav	3	angry	4	fear
Hindi_20.wav	3	angry	3	angry
Hindi_24.wav	3	angry	3	angry
Hindi_26.wav	3	angry	3	angry
Hindi_93.wav	4	fear	4	fear
Hindi_98.wav	4	fear	4	fear
Hindi_108.wav	4	fear	3	angry
Hindi_109.wav	4	fear	3	angry
Hindi_114.wav	4	fear	4	fear
Hindi_132.wav	1	happy	2	sad
Hindi_134.wav	1	happy	3	angry
Hindi_138.wav	1	happy	1	happy
Hindi_139.wav	1	happy	1	happy
Hindi_149.wav	1	happy	4	fear
Hindi_89.wav	0	neutral	0	neutral
Hindi_96.wav	0	neutral	0	neutral
Hindi_105.wav	0	neutral	1	happy
Hindi_116.wav	0	neutral	0	neutral
Hindi_119.wav	0	neutral	2	sad
Hindi_32.wav	2	sad	0	neutral
Hindi_36.wav	2	sad	2	sad
Hindi_47.wav	2	sad	2	sad
Hindi_61.wav	2	sad	4	fear
Hindi_63.wav	2	sad	2	sad

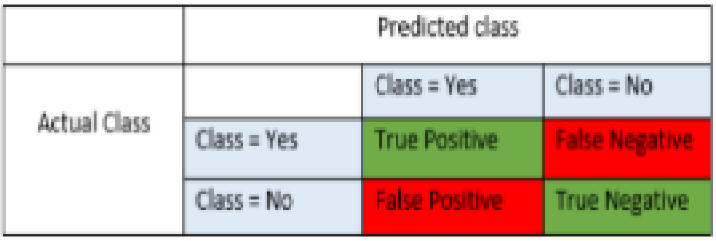
Table 5.4: Tested samples - Hindi

Sample name	Actual		Predicted	
	Value	Emotion	Value	Emotion
Malayalam_24.wav	3	angry	4	fear
Malayalam_26.wav	3	angry	3	angry
Malayalam_27.wav	3	angry	3	angry
Malayalam_28.wav	3	angry	3	angry
Malayalam_29.wav	3	angry	3	angry
Malayalam_6.wav	4	fear	4	fear
Malayalam_12.wav	4	fear	4	fear
Malayalam_14.wav	4	fear	4	fear
Malayalam_20.wav	4	fear	3	angry
Malayalam_38.wav	4	fear	3	angry
Malayalam_49.wav	1	happy	1	happy
Malayalam_50.wav	1	happy	1	happy
Malayalam_51.wav	1	happy	2	sad
Malayalam_53.wav	1	happy	3	angry
Malayalam_92.wav	1	happy	3	angry
Malayalam_43.wav	0	neutral	1	happy
Malayalam_45.wav	0	neutral	0	neutral
Malayalam_47.wav	0	neutral	0	neutral
Malayalam_52.wav	0	neutral	0	neutral
Malayalam_57.wav	0	neutral	2	sad
Malayalam_17.wav	2	sad	4	fear
Malayalam_21.wav	2	sad	2	sad
Malayalam_22.wav	2	sad	0	neutral
Malayalam_32.wav	2	sad	2	sad
Malayalam_34.wav	2	sad	3	angry

Table 5.5: Tested samples - Malayalam

5.2 PERFORMANCE EVALUATION

The most important characteristic of machine learning models is its ability to improve. Once the model is built, even before testing the model on real data, evaluation metrics reveal important model parameters and provides numeric scores that will help judge the functioning of the model. The most important metric needed to evaluate the model is the confusion matrix. The structure of a confusion matrix is against the actual and predicted positive and negative classes, and contains four values which are used to compute other metrics. The true positive represents the correct predictions made in the positive class, and the true negatives represent the correct predictions made in the negative class. The false positives and false negatives are the observations wrongly predicted for their respective classes. Validation is done on the datasets. Following figure 5.1 shows the evaluation parameters.



	Predicted class		
Actual Class		Class = Yes	Class = No
	Class = Yes	True Positive	False Negative
	Class = No	False Positive	True Negative

Figure 5.1: Evaluation metrics

Four important metrics can be derived using the values in the confusion matrix, namely:

$$Precision = \frac{Truepositives}{Truepositives + Falsepositives} \tag{5.1}$$

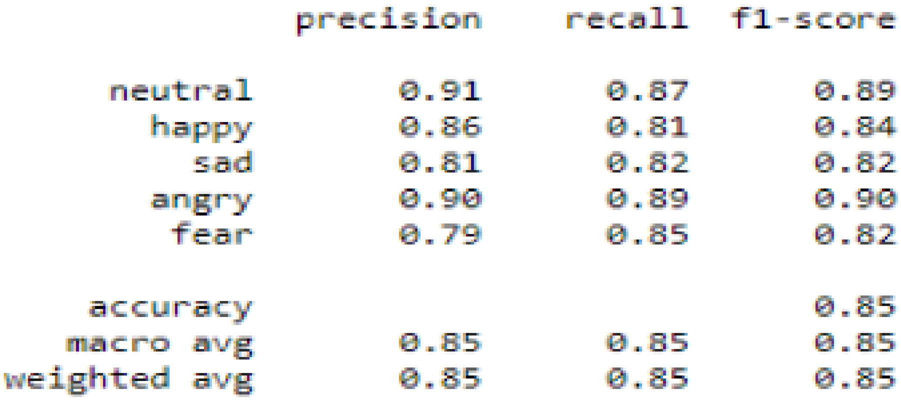
$$Recall = \frac{Truepositives}{Truepositives + Falsenegatives} \tag{5.2}$$

$$F1 = \frac{(2 * precision * recall)}{precision + recall} \tag{5.3}$$

$$Accuracy = \frac{Truepositives + Truenegatives}{AllTrueandFalsevalues} \tag{5.4}$$

5.2.1 Evaluation on dataset

Figure 5.3 shows the confusion matrix obtained on testing the dataset which was 20% of the whole dataset. Figure 5.2 shows the performance metrics obtained on testing the dataset.



```
              precision    recall  f1-score

     neutral       0.91      0.87      0.89
       happy       0.86      0.81      0.84
         sad       0.81      0.82      0.82
       angry       0.90      0.89      0.90
        fear       0.79      0.85      0.82

    accuracy                           0.85
   macro avg       0.85      0.85      0.85
weighted avg       0.85      0.85      0.85
```

Figure 5.2: Performance metrics - Testing dataset

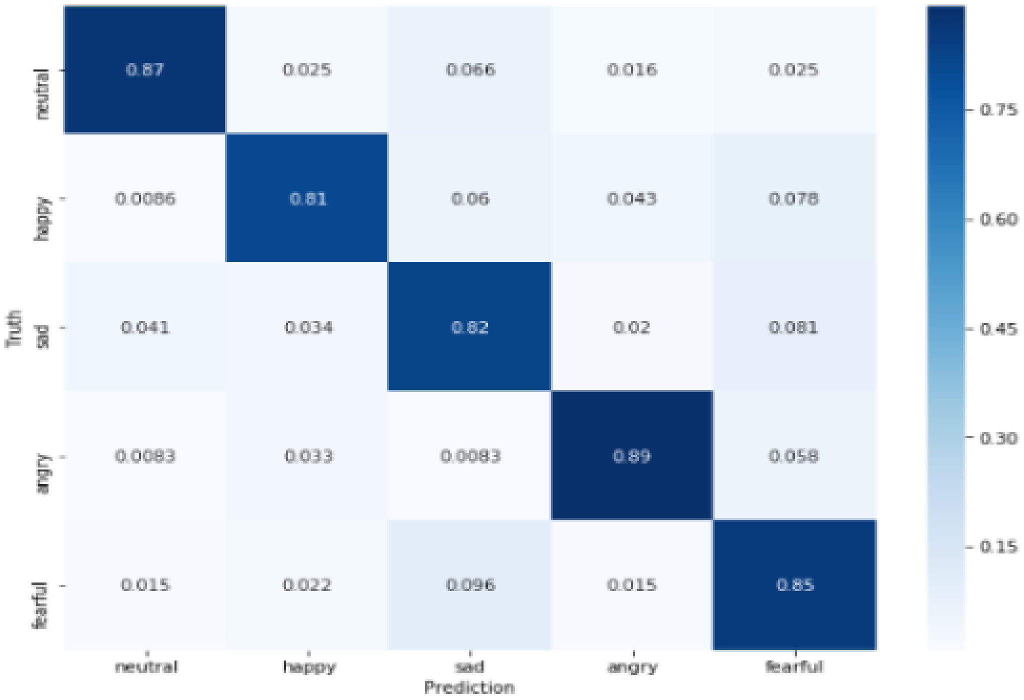
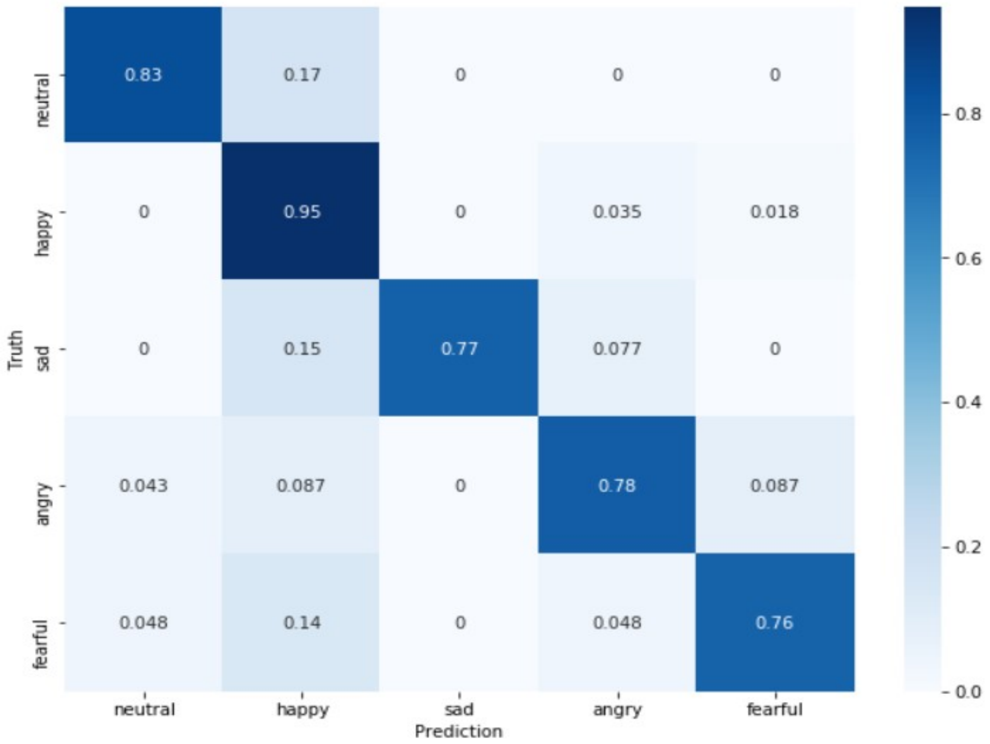


Figure 5.3: Confusion matrix - Testing dataset

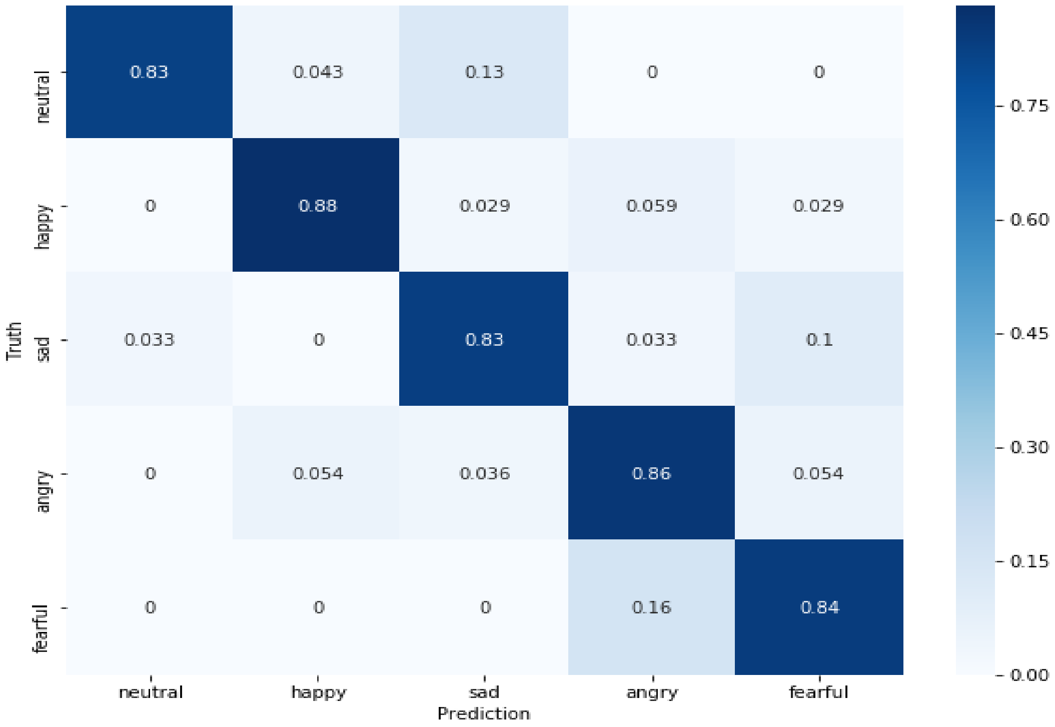
5.2.2 Evaluation on movie samples

Figure 5.4 lists the confusion matrix obtained for testing English audio samples from dataset. Figure 5.5, 5.6, 5.7 lists the confusion matrix obtained for testing manually collected Hindi, Tamil and Malayalam audio samples from movies. Overall confusion matrix for testing the audio samples is shown in figure 5.8



	neutral	happy	sad	angry	fearful
neutral	0.83	0.17	0	0	0
happy	0	0.95	0	0.035	0.018
sad	0	0.15	0.77	0.077	0
angry	0.043	0.087	0	0.78	0.087
fearful	0.048	0.14	0	0.048	0.76

Figure 5.4: Confusion matrix - English



	neutral	happy	sad	angry	fearful
neutral	0.83	0.043	0.13	0	0
happy	0	0.88	0.029	0.059	0.029
sad	0.033	0	0.83	0.033	0.1
angry	0	0.054	0.036	0.86	0.054
fearful	0	0	0	0.16	0.84

Figure 5.5: Confusion matrix - Hindi

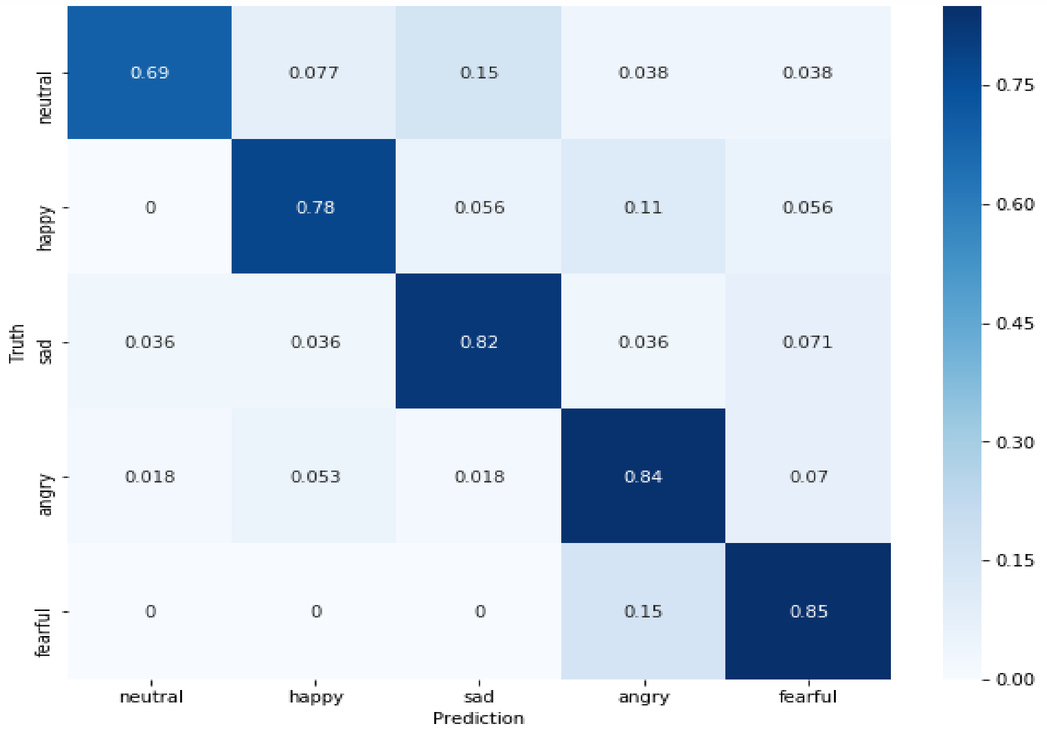


Figure 5.6: Confusion matrix - Tamil

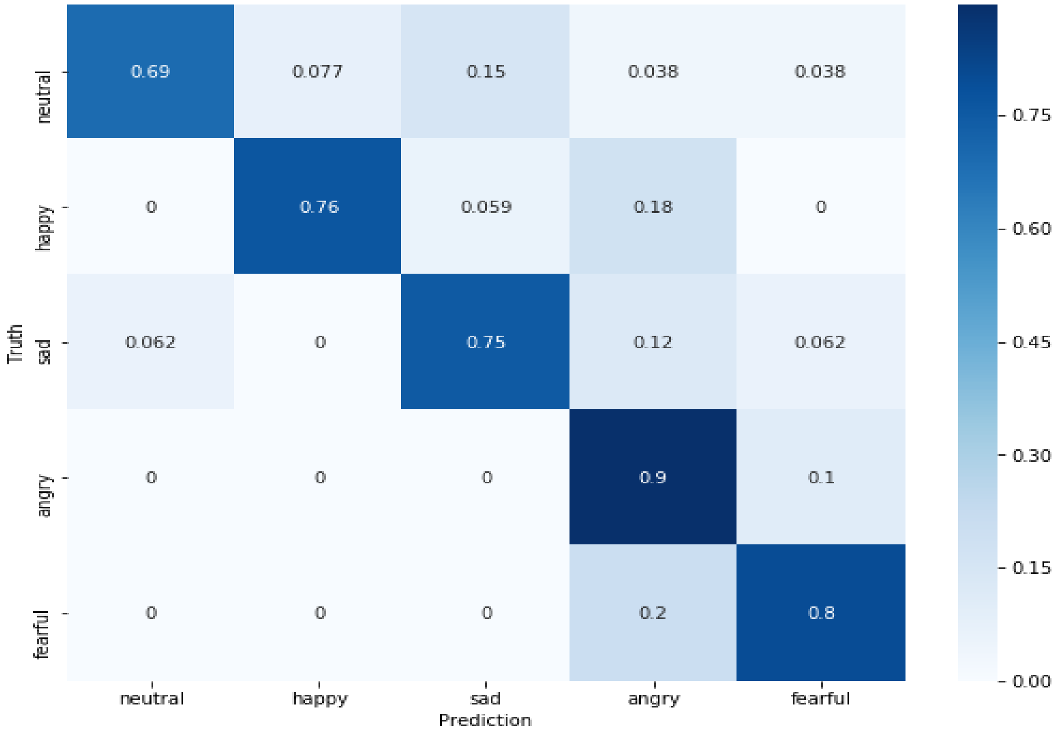


Figure 5.7: Confusion matrix - Malayalam

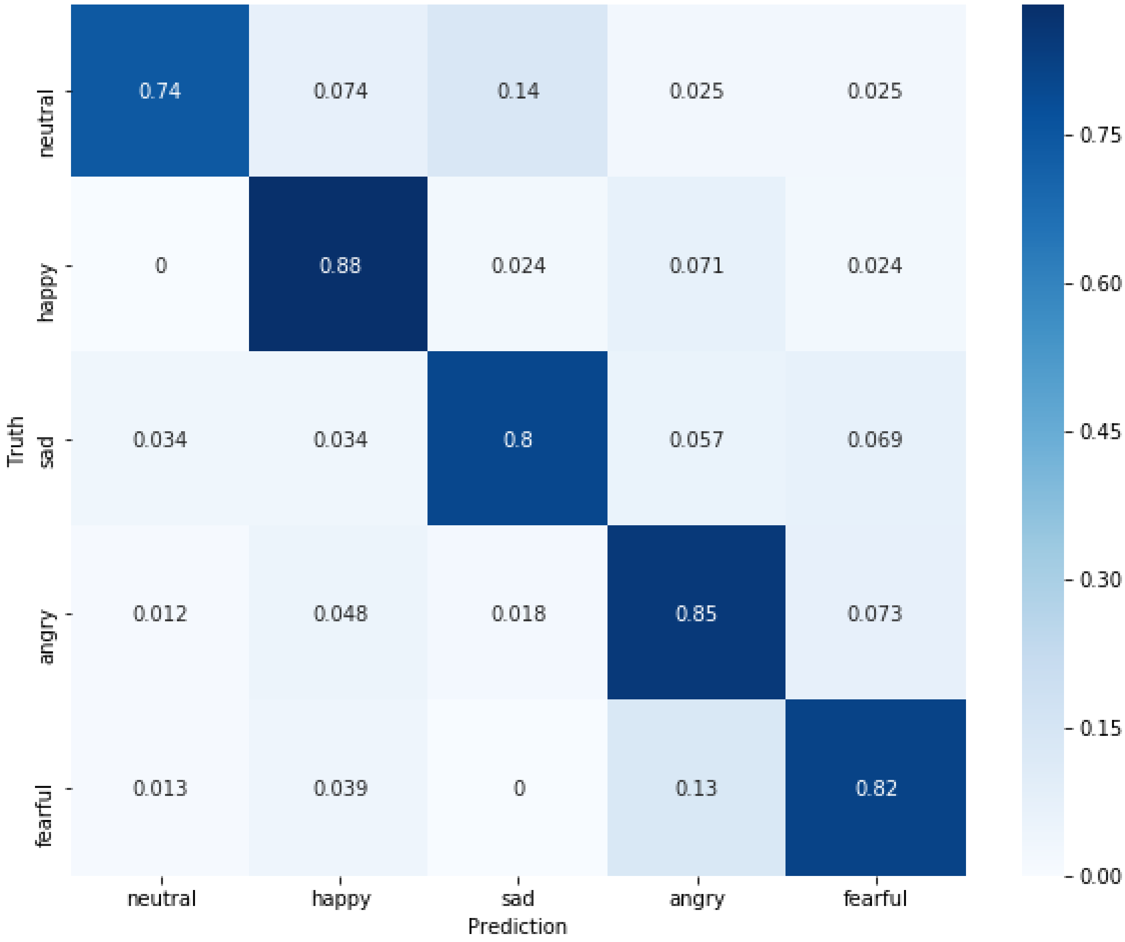
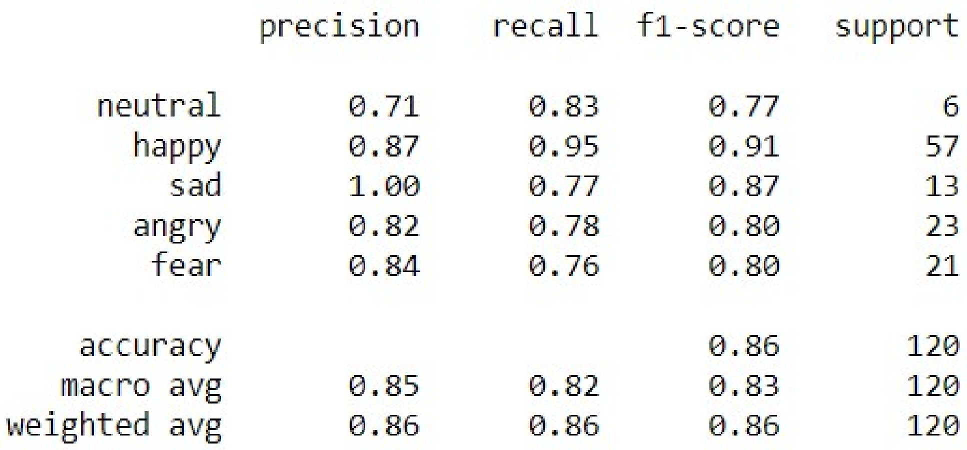


Figure 5.8: Overall Confusion matrix

The precision, recall and f-measure of English samples are 100%, 88% and 93% respectively and for Hindi samples are 100%, 88% and 93% respectively and for Tamil samples are 100%, 88% and 93% respectively and for Malayalam samples are 100%, 88% and 93% during testing. Hence, the accuracy of emotion recognition is 83% which is shown in figure 5.13. Figures 5.9, 5.10, 5.11 and 5.12 show the performances (in terms of Percentage) for English, Tamil, Hindi and Malayalam audio samples respectively. The X-axis denotes the performance metrics whereas y-axis denotes the obtained values in percentage.

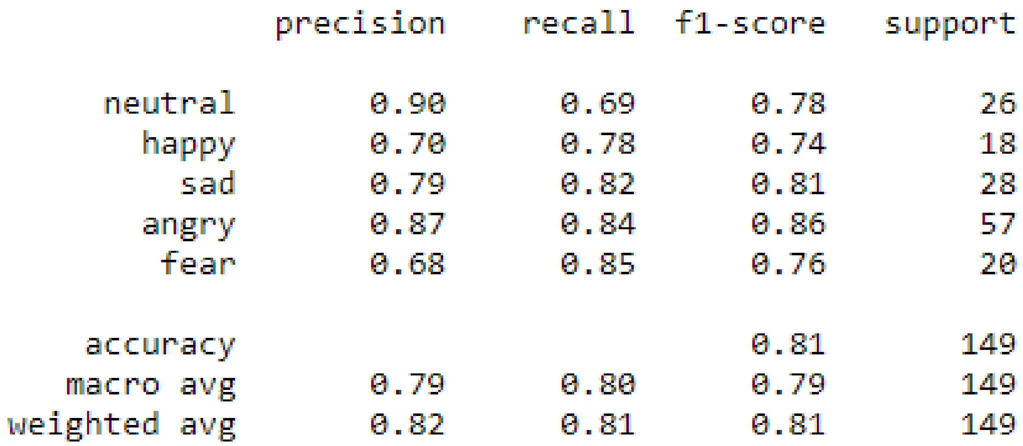


```
              precision    recall  f1-score   support

     neutral       0.71      0.83      0.77         6
       happy       0.87      0.95      0.91        57
         sad       1.00      0.77      0.87        13
       angry       0.82      0.78      0.80        23
        fear       0.84      0.76      0.80        21

    accuracy                           0.86       120
   macro avg       0.85      0.82      0.83       120
weighted avg       0.86      0.86      0.86       120
```

Figure 5.9: Performance metrics - English

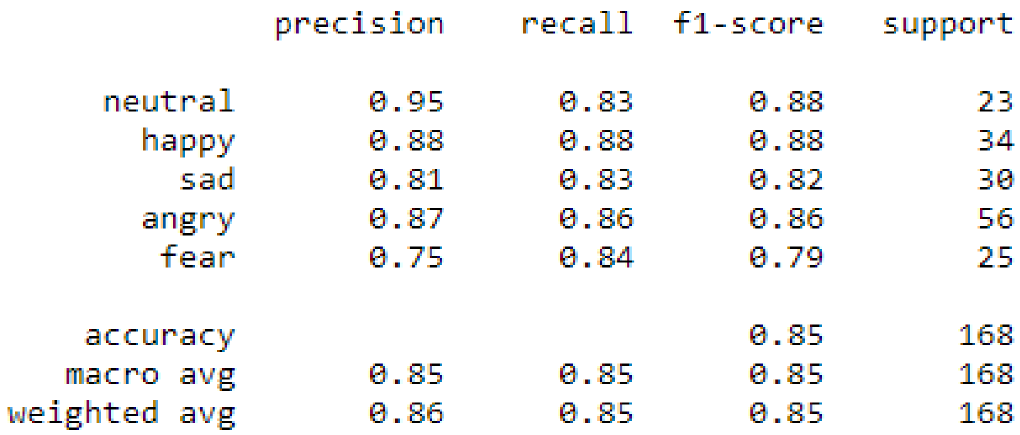


```
              precision    recall  f1-score   support

     neutral       0.90      0.69      0.78        26
       happy       0.70      0.78      0.74        18
         sad       0.79      0.82      0.81        28
       angry       0.87      0.84      0.86        57
        fear       0.68      0.85      0.76        20

    accuracy                           0.81       149
   macro avg       0.79      0.80      0.79       149
weighted avg       0.82      0.81      0.81       149
```

Figure 5.10: Performance metrics - Tamil

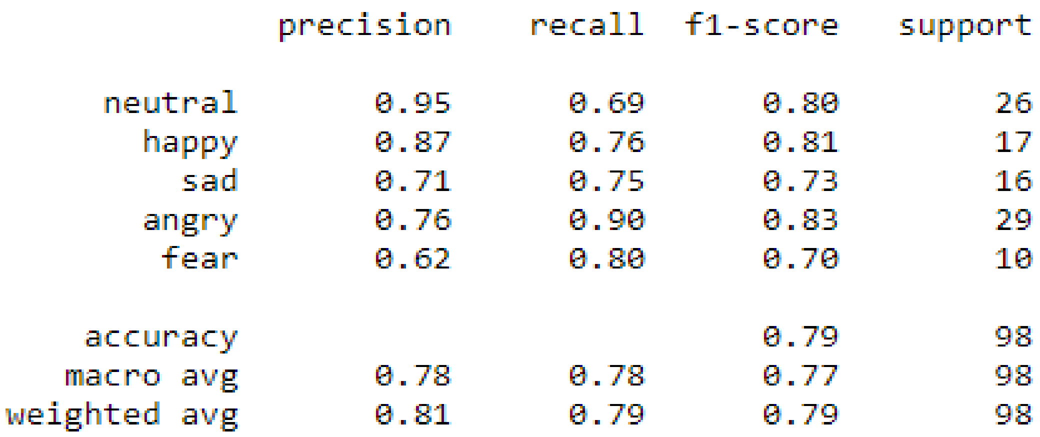


```
              precision    recall  f1-score   support

     neutral       0.95      0.83      0.88        23
       happy       0.88      0.88      0.88        34
         sad       0.81      0.83      0.82        30
       angry       0.87      0.86      0.86        56
        fear       0.75      0.84      0.79        25

    accuracy                           0.85       168
   macro avg       0.85      0.85      0.85       168
weighted avg       0.86      0.85      0.85       168
```

Figure 5.11: Performance metrics - Hindi

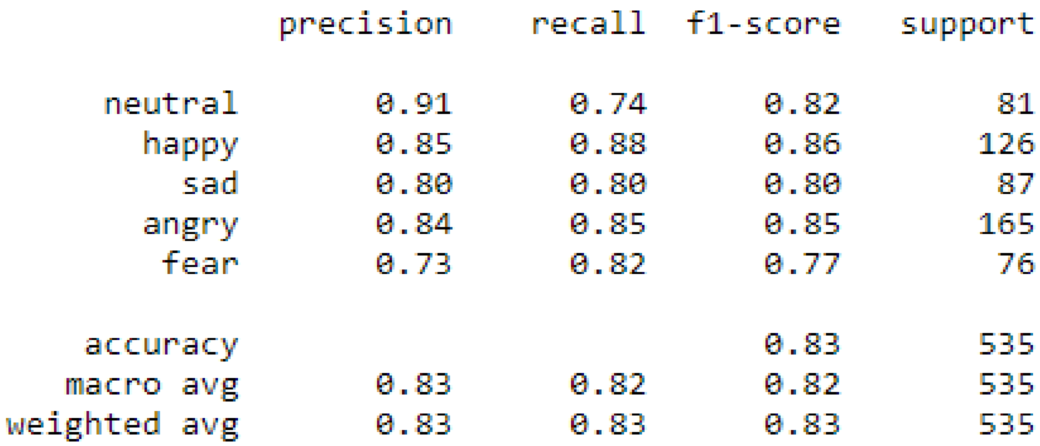


```
              precision    recall  f1-score   support

     neutral       0.95      0.69      0.80        26
       happy       0.87      0.76      0.81        17
         sad       0.71      0.75      0.73        16
       angry       0.76      0.90      0.83        29
        fear       0.62      0.80      0.70        10

    accuracy                           0.79        98
   macro avg       0.78      0.78      0.77        98
weighted avg       0.81      0.79      0.79        98
```

Figure 5.12: Performance metrics - Malayalam



```
              precision    recall  f1-score   support

     neutral       0.91      0.74      0.82        81
       happy       0.85      0.88      0.86       126
         sad       0.80      0.80      0.80        87
       angry       0.84      0.85      0.85       165
        fear       0.73      0.82      0.77        76

    accuracy                           0.83       535
   macro avg       0.83      0.82      0.82       535
weighted avg       0.83      0.83      0.83       535
```

Figure 5.13: Overall Performance metrics

Training the model with a single dataset yield an accuracy of 72%. So a combination of varying datasets are used which improved the accuracy to 98% around 60th epoch with a very minimal train loss of 4%. This also helped to increase the testing accuracy accordingly from 60% to 85%.

CHAPTER 6

CONCLUSION

6.1 SUMMARY

Our system is an efficient emotion recognition system which records the human speech, converts raw PCM audio data into .wav format, preprocess it for noise removal, separates voiced regions and feed the extracted spectral features include Mel Cepstral coefficients with 13 energy coefficients using python libraries into 4 layer 2 Dimensional Convolution neural network (4 layer 2D CNN) to train the speech signals and predict the emotion which falls under any of the five classes include Happy, sad, angry, fear and calm in real time. The model is deployed into android application bundle using chaquopy for real time identification of human emotion. This system recognizes emotion irrespective of languages and gender.

As there is an unavailability of human conversational speech datasets with different levels of emotions in different languages, audio samples are collected through various resources. Popular English datasets such as RAVDESS, SAVEE and TESS are used for training. The model accuracy is about 85% for the test data. English, Tamil, Hindi and Malayalam samples are collected from movies for evaluating performance in real-time environment.

The results of performance evaluation are very encouraging and shows that the accuracy of the system is about 83%. The results of confusion matrix and f-measure reveals that the emotion recognition in English, Tamil, Malayalam and Hindi is 86%, 81%, 79% and 85% efficient respectively.

Movie samples from various languages and live recorded speech from different actors in different languages are used to cross-validate the results. Distinct differences are observed between high-arousal emotions (Anger and Happy) and Neutral and Sad emotions. Results indicate overlap between Sad and Neutral emotions. But distinct differences are observed in the features for Happy/Anger and Fear, and between Happy and Anger emotions which is otherwise a challenging problem. The insights gained may be helpful in range of applications.

The system would work only for PCM and WAV audio files. Only five classes of emotions - Happy, sad, angry, fear and calm are trained for recognition. Integrating python with android to extract the features from audio signal in real time while recording through app ”Finding emo”, store audio file in device storage and recognize emotion was challenging. More audio samples under different situations and languages could improve the model for better results and avoid overlapping in predicting the emotions.

6.2 FUTURE WORK

The automatic recognition of human emotions would improve human computer interactions. Future work is to further explore the speech signals under various situations and in many more different languages. Such analyses would offer ascent to different applications that could enhance emotion recognition. Also, we have planned to detect the emotion for certain time frame say 2 seconds in each speech signal if time duration is more than 2 seconds and then recognize emotion based on the probability of the emotions recognized in each time frame. Besides the features in this proposed system where only 5 classes of emotions are trained for recognition, additional samples for various emotions like surprise, disgust, joy, cry etc., should be included. Emotions under various situations like crying in happiness, expressing sadness while laughing, shouting in emergency situations etc., should be included for better prediction and aftermath activities. Future work includes publishing the android application in play store.

REFERENCES

[1] Mustaqeem . and Soonil Kwon, “A cnn-assisted enhanced audio signal processing for speech emotion recognition”, *Sensors*, vol. 20, p. 183, 12 2019.

[2] Paul Boersma and David Weenink, “Praat: Doing phonetics by computer (version 6.0.37)”, 03 2018.

[3] Philip Jackson and Sana ul haq, “Surrey audio-visual expressed emotion (savee) database”, 04 2011.

[4] Yannick Jadoul, Bill Thompson, and Bart de Boer, “Introducing parselmouth: A python interface to praat”, *Journal of Phonetics*, vol. 91, pp. 1–15, 11 2018.

[5] R. A. Khalil, E. Jones, M. I. Babar, T. Jan, M. H. Zafar, and T. Alhussain, “Speech emotion recognition using deep learning techniques: A review”, *IEEE Access*, vol. 7, pp. 117327–117345, 2019.

[6] M. S. Likitha, S. R. R. Gupta, K. Hasitha, and A. U. Raju, “Speech based human emotion recognition using mfcc”, In *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, pp. 2257–2260, 2017.

[7] Steven R Livingstone and Frank A. Russo, “The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english”, In *PloS one*, 2018.

[8] Sneha Lukose and Savitha Upadhya, “Music player based on emotion recognition of voice signals”, pp. 1751–1754, 07 2017.

[9] Sona Patel, Klaus Scherer, Johan Sundberg, and Eva Björkner, “Acoustic markers of emotions based on voice physiology”, volume 100865, 01 2010.

[10] M. Kathleen Pichora-Fuller and Kate Dupuis, “Toronto emotional speech set (TESS)”, 2020.

[11] Rashmirekha Ram, Hemanta Palo, and Narayan Mohanty, “Emotion recognition with speech for call centres using lpc and spectral analysis”, *International Journal of Advanced Computer Research*, vol. Volume-3 September-2013, pp. 189–194, 09 2013.

[12] E. Ramdinmawii, A. Mohanta, and V. K. Mittal, “Emotion recognition from speech signal”, In *TENCON 2017 - 2017 IEEE Region 10 Conference*, pp. 1562–1567, 2017.

[13] Y. Xie, R. Liang, Z. Liang, C. Huang, C. Zou, and B. Schuller, “Speech emotion classification using attention-based lstm”, *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, num. 11, pp. 1675–1685, 2019.