DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

Rajiv Gandhi University of Knowledge Technologies – Nuzvid

Nuzvid, Eluru, Andhra Pradesh – 521202.

VOICE EMOTION DETECTION MODEL

A Project Progress Report

Submitted in partial fulfillment for the degree of

BACHELOR OF TECHNOLOGY
In

COMPUTER SCIENCE AND ENGINEERING

Submitted by

B. Idris Baig (N180726)

D.Karthik (N180241)

Md.Althaf (N180253)

S.Manjulatha (N180490)

K.Neeraja (N180530)

Under the Esteem Guidance of

Mrs. Medisetty Baby Anusha

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

Rajiv Gandhi University of Knowledge Technologies – Nuzvid

Nuzvid, Eluru, Andhra Pradesh – 521202.

CERTIFICATE OF COMPLETION

This is to certify that the work entitled, "Voice Emotion Detection using LSTM model" is the bonafied work of **B.IDRIS BAIG (IDNo: N180726), D.KARTHIK (ID No:N180241), MD.ALTHAF (ID No: N180253), S.MANJULATHA (IDNo:N180490), K.NEERAJA (ID No:N180530)** carried out under my guidance and supervision for 3rd year mini project of Bachelor of Technology in the department of Computer Science and Engineering under RGUKT IIIT, Nuzvid. This work is done during the academic session February 2023 – May 2023, under our guidance.

Mrs.Medisetty Baby Anusha
Assistant professor,
Department of CSE,
RGUKT, Nuzvid

Mr.Chiranjeevi Sadu
Assistant Professor,
Head of the Department,
Department of CSE,
RGUKT, Nuzvid.

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

Rajiv Gandhi University of Knowledge Technologies – Nuzvid

Nuzvid, Eluru, Andhra Pradesh – 521202.

CERTIFICATE OF EXAMINATION

This is to certify that the work entitled, "Voice Emotion Detection using LSTM model" is the bonafied work of **B.IDRIS BAIG (IDNo: N180726), D.KARTHIK (ID No: N180241), MD.ALTHAF (ID No: N180253), S.MANJULATHA (IDNo:N180490), K.NEERAJA (ID No:N180530)** and here by accord our approval of it as a study carried out and presented in a manner required for its acceptance in 3rd year of Bachelor Of Technology for which it has been submitted. This approval does not necessarily endorse or accept every statement made, opinion expressed or conclusion drawn, as recorded in this thesis. It only signifies the acceptance of this thesis for the purpose for which it has been submitted.

Mrs.Medisetty Baby Anusha

Assistant Professor,

Department of CSE,

RGUKT-Nuzvid.

Project Examiner

RGUKT-Nuzvid.

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

Rajiv Gandhi University of Knowledge Technologies – Nuzvid

Nuzvid, Eluru, Andhra Pradesh – 521202.

DECLARATION

We " **B.IDRIS BAIG (IDNo: N180726), D.KARTHIK (ID No: N180241), MD.ALTHAF (ID No: N180253), S.MANJULATHA (IDNo:N180490), K.NEERAJA (ID No:N180530)**)"hereby declare that the project report entitled "Voice Emotion Detection using LSTM model" done by us under the guidance of Mrs.Medisetty Baby Anusha, Assistant Professor, is submitted for the fulfillment of mini project during the academic session February 2023- June 2023 at RGUKT-Nuzvid. We also declare that this project is a result of our own effort and has not been copied or imitated from any source. Citations from any websites are mentioned in the references. The results embodied in this project report have not been submitted to any other university or institute for the award of any degree or diploma.

Date:12-06-2023

Place: Nuzvid

B.IDRIS BAIG (N180726)
D.KARTHIK (N180241)
MD.ALTHAF (N180253)
S.MANJULATHA (N180490)
K.NEERAJA (N180530)

ACKNOWLEDGMENT

I would like to express my gratitude towards my advisor **Medisetty Baby Anusha** mam for guiding me through the extensive research process.We shall always cherish the time spent with him during the course of this work due to the invaluable knowledge gained in the field of reliability engineering.

We are extremely grateful for the confidence bestowed in us and entrusting our project entitled“Voice Emotion Detection”. We express our gratitude to Mrs. Medisetty Baby Anusha(Asst prof of CSE) and other faculty members for being source of inspiration and constant encouragement which helped us in completing the project successfully.

Finally, yet importantly, we would like to express our heartfelt thanks to our beloved God and parents for their blessings, our friends for their help and wishes for the successful completion of this project.

ABSTRACT

Communication is the key to express one's thoughts and ideas clearly. Amongst all forms of communication, speech is the most preferred and powerful form of communication in humans. The era of the Internet of Things (IoT) is rapidly advancing in bringing more intelligent systems available for everyday use. These applications range from simple wearables and widgets to complex self-driving vehicles and automated systems employed in various fields. Intelligent applications are interactive and require minimum user effort to function, and mostly function on voice-based input.

This creates the necessity for these computer applications to completely comprehend human speech. A speech percept can reveal information about the speaker including gender, age, language, and emotion. Several existing speech recognition systems used in IoT applications are integrated with an emotion detection system in order to analyze the emotional state of the speaker. The performance of the emotion detection system can greatly influence the overall performance of the IoT application in many ways and can provide many advantages over the functionalities of these applications. This research presents a speech emotion detection system with improvements over an existing system in terms of data, feature selection, and methodology that aims at classifying speech percepts based on emotions, more accurately.

TABLE OF CONTENTS

CHAPTER -1

INTRODUCTION

For several years now, the growth in the field of Artificial Intelligence (AI) has been accelerated. AI, which was once a subject understood by computer scientists only, has now reached the house of a common man in the form of intelligent systems. The advancements of AI have engendered several technologies involving Human-Computer Interaction (HCI). Aiming to develop and improve HCI methods is of paramount importance because HCI is the front-end of AI which millions of users experience. Some of the existing HCI methods involve communication through touch, movement, hand gestures, voice and facial gestures. Among the different methods, the voice-based intelligent devices are gaining popularity in a wide range of applications. In a voice-based system, a computer agent is required to completely comprehend the human's speech percept in order to accurately pick up the commands given to it. This field of study is termed as Speech Processing and consists of three components: Speaker Identification Speech Recognition Speech Emotion Detection Speech Emotion Detection is challenging to implement among the other components due to its complexity. Furthermore, the definition of an intelligent computer system requires the system to mimic human behavior. A striking nature unique to humans is the ability to alter conversations based on the emotional state of the speaker and the listener. Speech emotion detection can be built as a classification problem solved using several machine learning algorithms. This project discusses in detail the various methods and experiments carried out as part of implementing a Speech Emotion Detection system.

1.1 IMPORTANCE

Communication is the key to expressing oneself. Humans use most parts of their body and voice to effectively communicate. Hand gestures, body language, and the tone and temperament are all collectively used to express one's feelings. Though the verbal part of the communication varies by languages practiced across the globe, the non-verbal part of communication is the expression of feeling which is most likely common among all. Therefore, any advanced technology developed to produce a social environment experience also covers understanding emotional context in speech. Improvements in the field of emotion detection positively impact a multitude of applications. Some of the research areas that benefit from automating the emotion detection technique include psychology, psychiatry, and neuroscience. These departments of cognitive sciences rely on human interaction, where the subject of study is put through a series of questions and situations, and based on their reactions and responses, several inferences are made. A potential drawback occurs as few people are classified introverts and hesitate to communicate. Therefore, replacing the traditional procedures with a computer-based detection system can benefit the study. Similarly, the practical applications of speech-based emotion detection are many. Smart home appliances and assistants (Examples: Amazon Alexa and Google Home are ubiquitous these days. Additionally, customer care-based call centers often have an automated voice control which might not please most of their angry customers. Redirecting such calls to a human attendant will improve the service. Other applications include eLearning, online tutoring, investigation, personal assistant (Example: Apple Siri and Samsung S Voice etc. A very recent application could be seen in self-driving cars. These vehicles heavily depend on voice-based controls. An unlikely situation, such as anxiety, can cause the passenger to utter unclear sentences. In these situations, understanding the emotional content expressed becomes of prime importance.

1.2 MOTIVATION

Identifying the emotion expressed in a speech percept has several use cases in the modern day applications. Human-Computer Interaction (HCI) is a field of research that studies interactive applications between humans and computers . For an effective HCI application, it is necessary for the computer system to understand more than just words. On the other hand, the field of Internet of Things (IoT) is rapidly growing. Many real world IoT applications that are used on a daily basis such as Amazon Alexa, Google Home and Mycroft function on voice-based inputs. The role of voice in IoT applications is pivotal. The study in a recent article foresees that by 2022, about 12% of all IoT applications would fully function based on voice commands only. These voice interactions could be mono-directional or bi-directional, and in both cases, it is highly important to comprehend the speech signal. Further, there are Artificial Intelligence (AI) and Natural Language Processing (NLP) based applications that use functions of IoT and HCI to create complex systems. Self-driving cars are one such application that controls many of its functions using voice-based commands. Identifying the emotional state of the user comes with a great advantage in this application. Considering emergency situations in which the user may be unable to clearly provide a voice command, the emotion expressed through the user's tone of voice can be used to turn on certain emergency features of the vehicle. A much simpler application of speech emotion detection can be seen in call centers, in which automated voice calls can be efficiently transferred to customer service agents for further discussion. Other applications of using a speech emotion detection system can be found in lie detecting systems, criminal department analysis, and in humanoids.

CHAPTER 2

PRELIMINARIES

This section describes the basics of the speech-emotion recognition system. This system inputs speech (audio) of various emotions. And gives a specific emotion based on the input data. Some terms are used for the system.

2.1 SPEECH(Audio)

Audio is a sound that has a frequency of 20 Hz to 20 K Hz. For the system, only human speeches are used. There are two types of audio: mono and stereo. Monoaudio is recorded with a single channel (microphone) and stereo is recorded with multiple channels . This system uses mono audio data. Audio data are in many formats such as .wav, .mp3, .mp4, .flac, .aac, .wma, etc. The datasets used here are in .wav format. WAV audio stores the waveform data. It is an uncompressed audio file. It is the strongest and has high-quality audio data.

2.2 EMOTION

Human emotions are the mental states which are bought by the neurophysiological changes of humans. It is based on thoughts, behavioral responses, feelings etc.
Various types of emotion are seen in human behavior such as happiness, surprise, sadness, fear, anger, disgust,neutrality etc. This system classified these emotions.

2.3 FEATURE EXTRACTION

Feature extraction is extracting necessary features from the input data for the target classification. For audio data, some techniques can be used for feature extraction.

The most used technique is MFCC. MFCCs are the Mel Frequency Cepstral Coefficients. It is one of the best spectral features . It converts conventional frequency into Mel Scale and for this MFCC is used . Mel scale is mainly a non-linear scale. For calculating mels, the equation is:

$$f(mel)=\{1000\log_{10}(1+f/1000)\}/\log 102. \quad (1)$$

By calculating mel following equation (1), MFCC extracted the necessary features. In this system, the MFCC technique is used for the task of feature extraction. Which provides the best result for speech emotion recognition.

2.4 SYSTEM REQUIREMENTS

I. Hardware Requirement

i. Laptop or PC

- Windows 8 or higher
- I3 processor system or higher
- 4 GB RAM or higher
- 100 GB ROM or higher

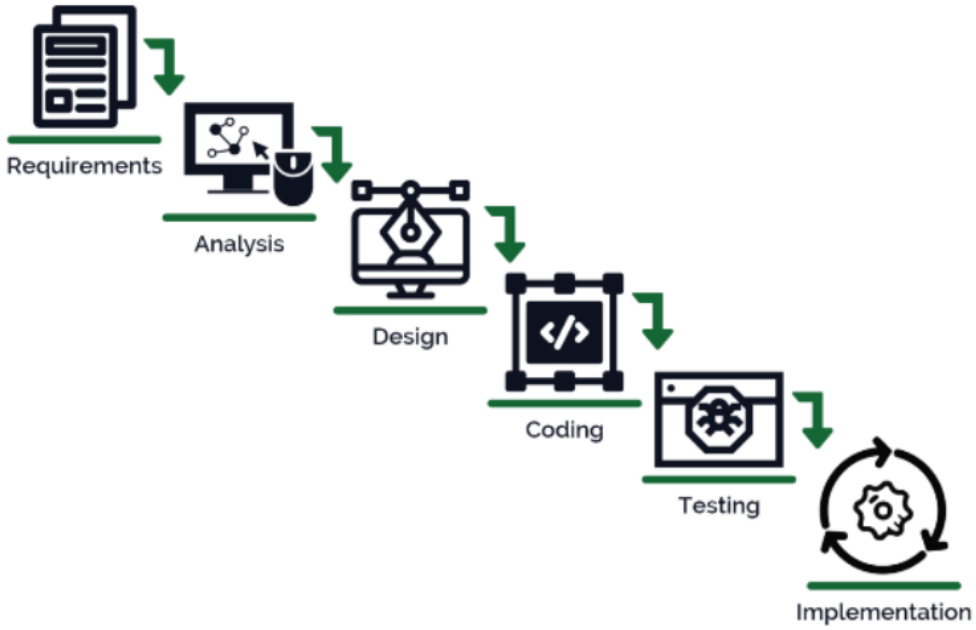
II. Software Requirement

ii. Laptop or PC

- Python
- Jupyter Notebook
- Google colab
- PyCharm

2.5 PROJECT LIFE CYCLE MODEL

The waterfall model is a classical model used in the system development life cycle to create a system with a linear and sequential approach. It is termed a waterfall because the model develops systematically from one phase to another in a downward fashion. The waterfall approach does not define the process to go back to the previous phase to handle changes in requirements. The waterfall approach is the earliest approach that was used for software development.

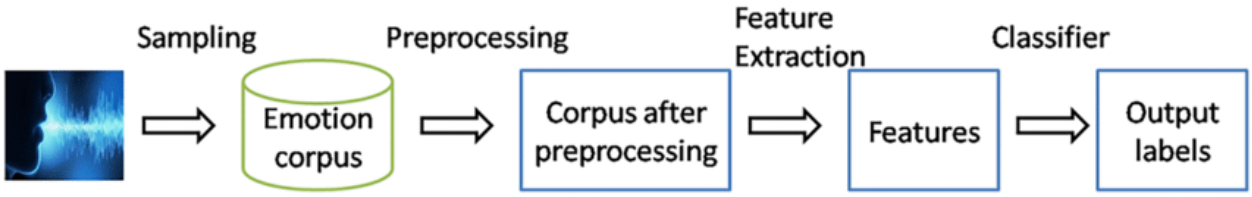


CHAPTER-3

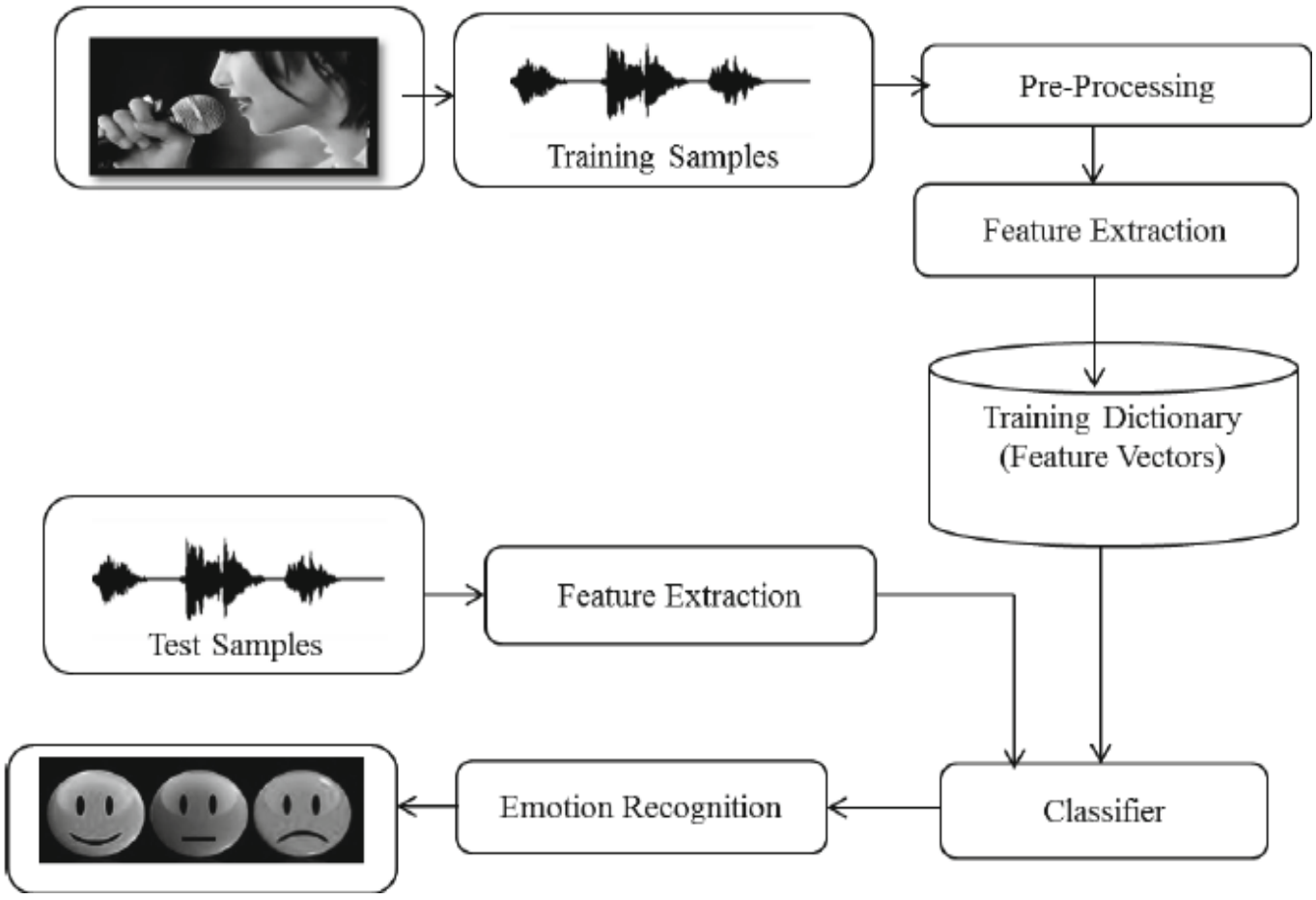
STRATEGY

3.1 METHODOLOGY

The speech emotion detection system is implemented as a Machine Learning (ML) model. The steps of implementation are comparable to any other ML project, with additional fine-tuning procedures to make the model function better. The flowchart represents a pictorial overview of the process . The first step is data collection, which is of prime importance. The model being developed will learn from the data provided to it and all the decisions and results that a developed model will produce is guided by the data. The second step, called feature engineering, is a collection of several machine learning tasks that are executed over the collected data. These procedures address the several data representation and data quality issues. The third step is often considered the core of an ML project where an algorithmic based model is developed. This model uses an ML algorithm to learn about the data and train itself to respond to any new data it is exposed to. The final step is to evaluate the functioning of the built model. Very often, developers repeat the steps of developing a model and evaluating it to compare the performance of different algorithms. Comparison results help to choose the appropriate ML algorithm most relevant to the problem.



3.2 ARCHITECTURE OF THE SYSTEM



3.3 DATASETS

Three types of speech datasets are used in this SER system. TESS, SAVEE, and RAVDESS are the most common audio dataset for speech emotion classification.

3.3.1 TESS Dataset

The Toronto Emotional Speech Set (TESS) provides us with high-quality speeches by female speakers . Most of the datasets are male only . But this female-only
the dataset gives a balance in audio classification. So, this dataset can train well and provides a good model (without overfitting). It contains at most 2800 audio files in WAV format . Two female speakers speak 200 target audio words . That means a total of 400 audio data is spoken for a single emotion . Seven emotions are in this dataset: fear, happiness, anger, disgust, neutral, surprise, and sadness .

3.3.2 RAVDESS Dataset

The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) is a well-known dataset for audio classification . 24 speakers speak in a total of 1440 audio speeches. 12 male and 12 female speakers speak of eight emotions here: happy, sad, calm, angry, surprised, disgusted, fearful, and neutral .Every speaker speaks 60 speeches and all are in WAV format . It also provides audio files with strong and normal emotional intensity . These three datasets are concatenated with each other as TESS+SAVEE, TESS+RAVDESS, andTESS+SAVEE+RAVDESS for increasing the number of training datasets and classifying speech emotions more accurately. So, these three datasets TESS + SAVEE,TESS + RAVDESS, and TESS + SAVEE + RAVDESS datasets are used for this experiment and training model.
In this experiment, 7 emotions are used: fear, happiness,neutral, anger, disgust, sadness, and surprise.

3.3.3 SAVEE Dataset

The Surrey Audio-Visual Expressed Emotion (SAVEE) is a high-quality audio dataset that is spoken by only male speakers . A total of 480 audio files are in the SAVEE dataset in WAV format. 120 audios are for neutral emotions and 60 files are for other emotions . Seven emotions are in this dataset: fear, happiness, anger, neutral, surprise, disgust, and sadness. It would be a good interaction with other male-only datasets.

3.4 PREPROCESSING

The dataset is preprocessed with the resample type of kaiser_fast, reducing the dataset's load time. It sampled with a sample rate of 44100. Data is split into training and testing parts. Here, 75% of the data is used for model training, and 25% is used for model testing. Data split is done by shuffling the dataset with the random state 42. This increases the performance of the model train. We used the 80:20 data split and the 75:25 data split. Where we got the best results for a 75:25 split of the data. To get a good model, it needs more data and tests with less data than train This work provided good results for this segmentation of 75% train data and 25% test data.

3.5 FEATURE EXTRACTION

Speech input needs to convert into digital signals to train the model . After converting into digital signals, the audio signal is processed, and extracted from the related suitable features for training the model . For extracting features, MFCCs are used in this system.

3.6 DATA NORMALIZATION

Primarily the audio data is converted to the array. Then mean and the standard deviation (SD) are found from the dataset. Then normalize the data by subtracting the mean and dividing the subtraction by the standard deviation (SD). Data is again converted to a numpy array and expanded dims for train CNN, RNN, CNN-LSTM, and LSTM models.

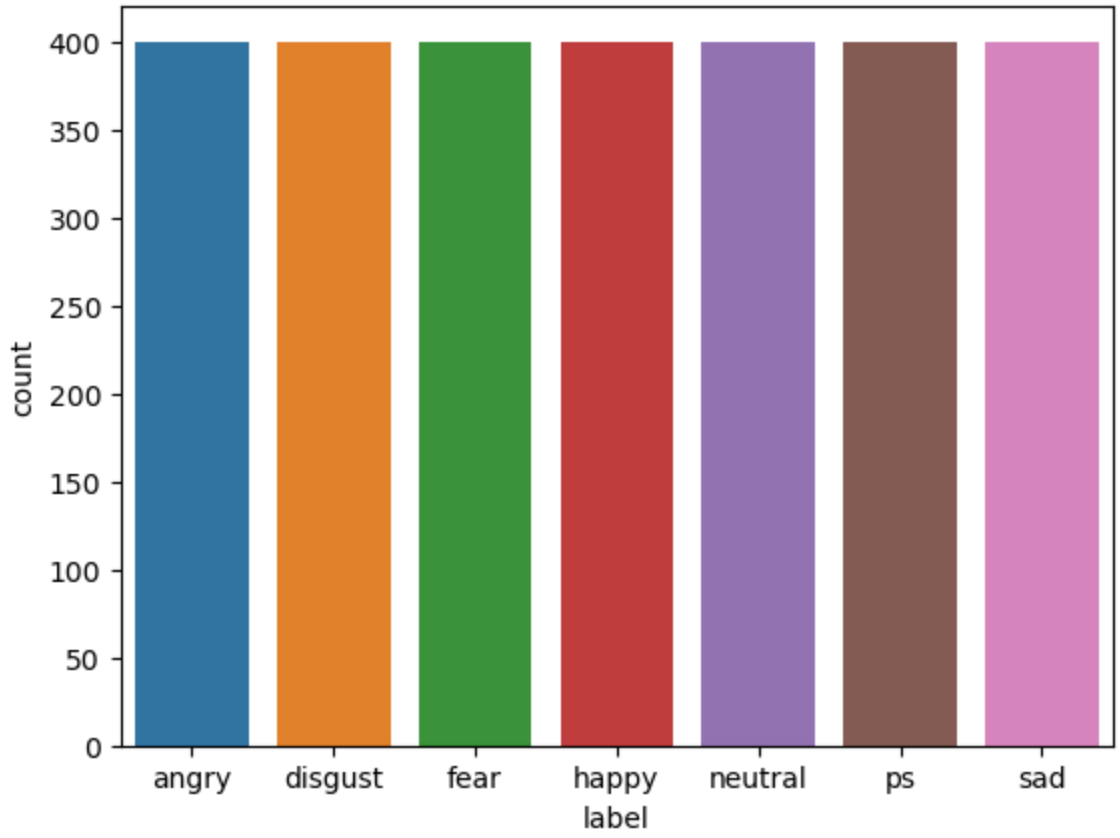
3.7 DATA VISUALIZATION

Data visualization is a graphical representation of quantitative information and data by using visual elements like graphs, charts, and maps. Data visualization converts large and small data sets into visuals, which is easy to understand and process for humans. Data visualization tools provide accessible ways to understand outliers, patterns, and trends in the data.

In the world of Big Data, the data visualization tools and technologies are required to analyze vast amounts of information. Data visualizations are common in your everyday life, but they always appear in the form of graphs and charts. The combination of multiple visualizations and bits of information are still referred to as Infographics. Data visualizations are used to discover unknown facts and trends. You can see visualizations in the form of line charts to display change over time. Bar and column charts are useful for observing relationships and making comparisons. A pie chart is a great way to show parts-of-a-whole. And maps are the best way to share geographical data visually.

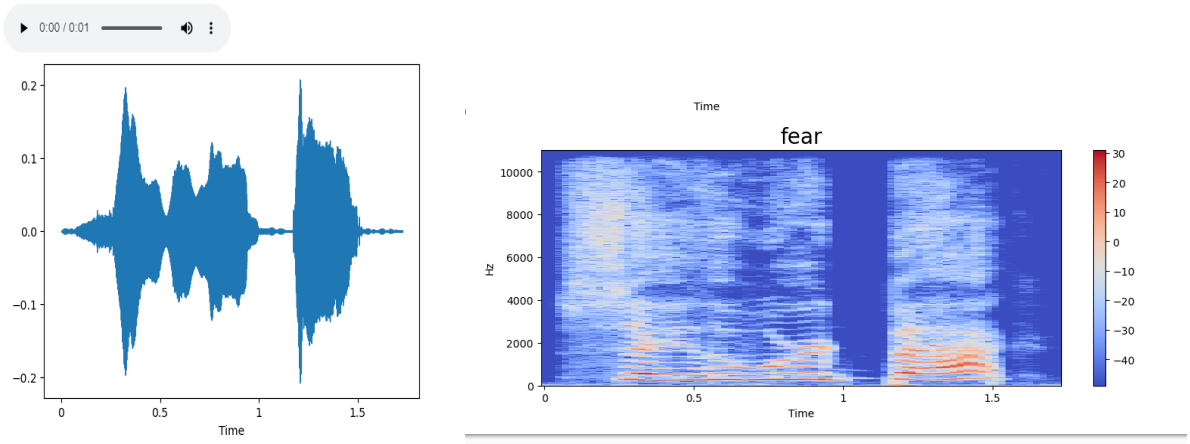
```
sns.countplot(df['label'])
```

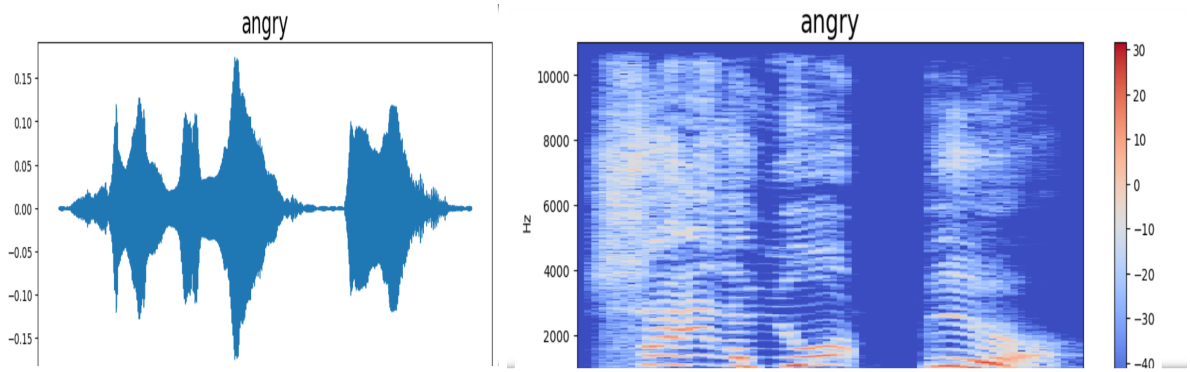
```
<AxesSubplot:xlabel='label', ylabel='count'>
```

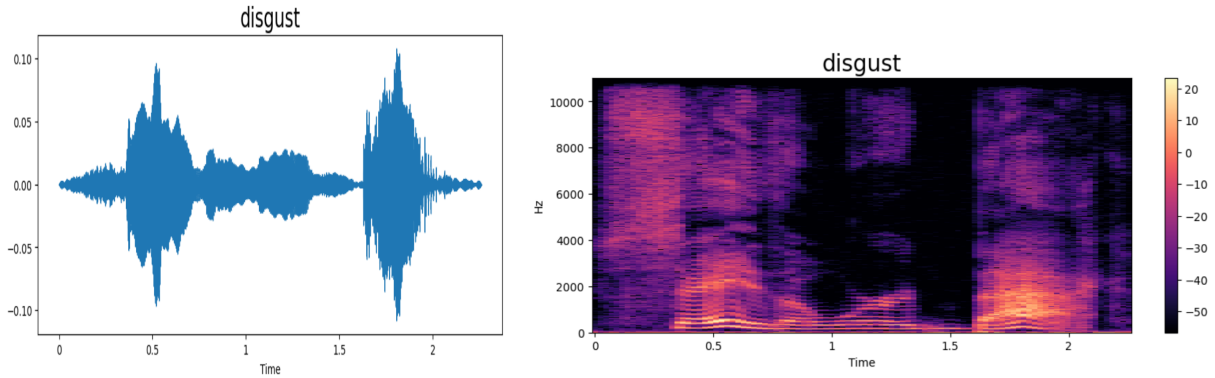


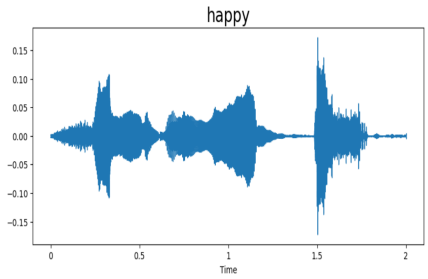

```
[ ]  emotion = 'fear'
     path = np.array(df['speech'][df['label']==emotion])[0]
     data, sampling_rate = librosa.load(path)
     librosa.display.waveshow(data)
     # librosa.display.waveshow(samplig_rate)
     spectogram(data,sampling_rate,emotion)
     Audio(path)
```

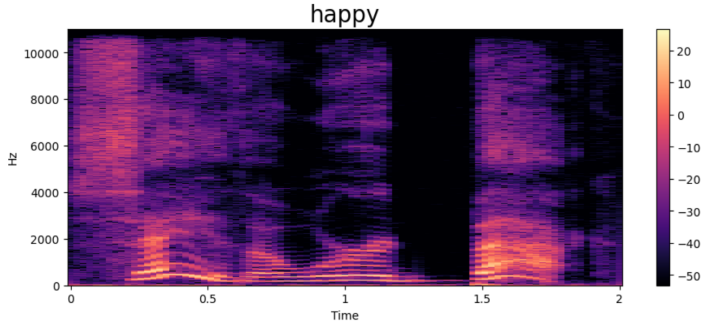
OUTPUT:

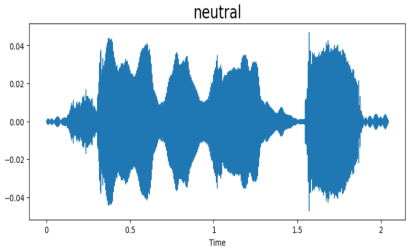

0:00 / 0:01
Time
fear
Hz
Time

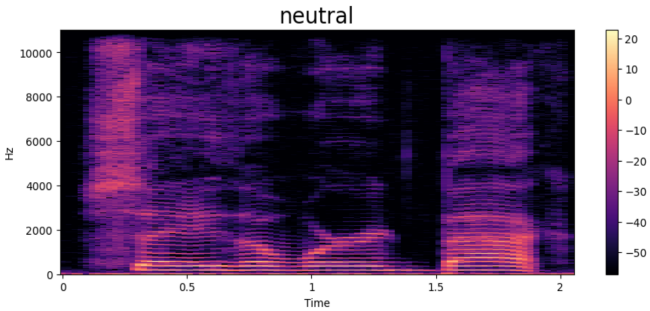

angry
angry
Hz

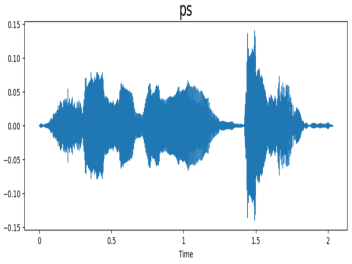

disgust
disgust
Hz
Time

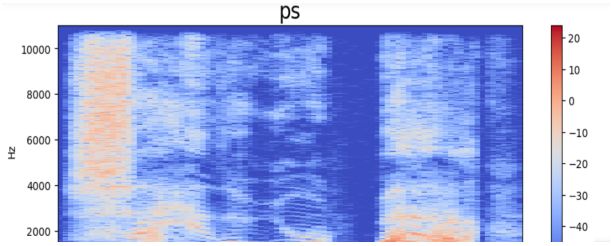

happy
Time

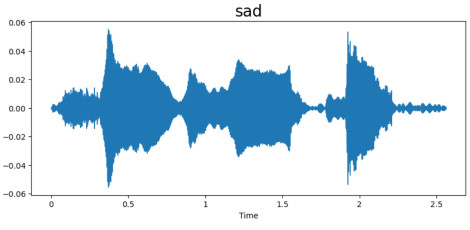

happy
Hz
Time

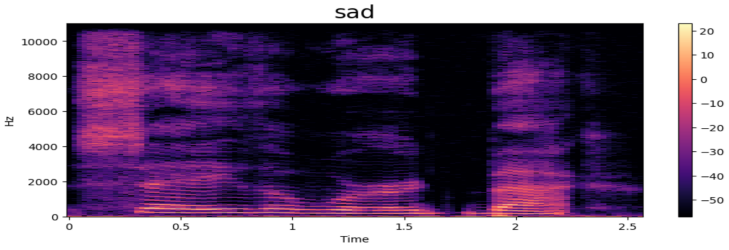

neutral
Time


neutral
Hz
Time


ps
Time


ps
Hz


sad
Time

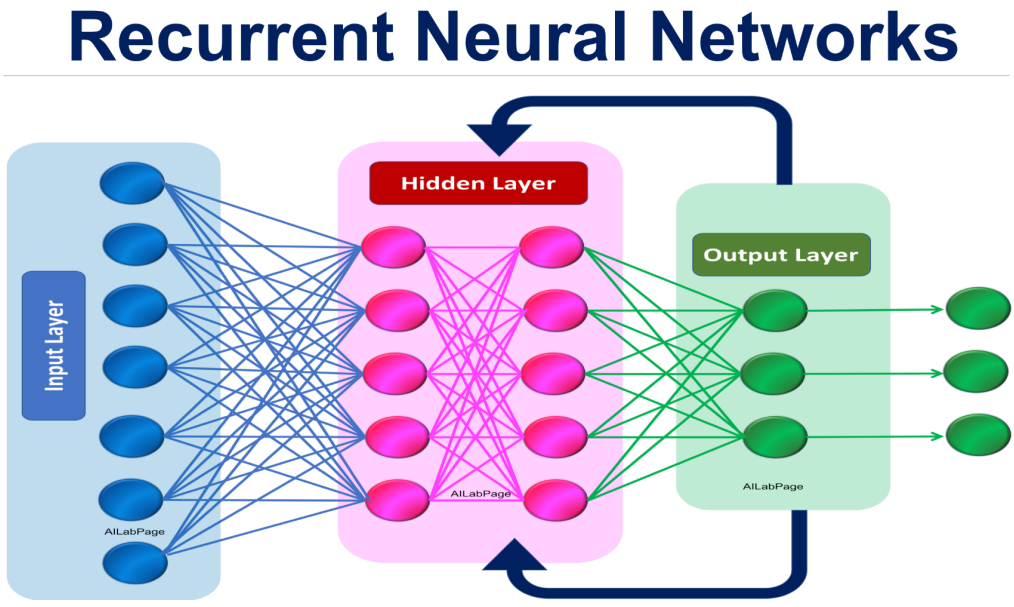

sad
Hz
Time

3.8 MODELS

In this system, 3 deep learning models are used. All 3 models are trained with all three datasets (TESS + SAVEE, TESS + RAVDESS, TESS + SAVEE + RAVDESS). The models are RNN, LSTM.

3.8.1 RNN

Recurrent Neural Network(RNN) is a type of Neural Network, where the **output from the previous step is fed as input to the current step**. In traditional neural networks, all the inputs and outputs are independent of each other, but in cases like when it is required to predict the next word of a sentence, the previous words are required and hence there is a need to remember the previous words. Thus RNN came into existence, which solved this issue with the help of a Hidden Layer. The main and most important feature of RNN is the Hidden **state**, which remembers some information about a sequence.



.

Pseudocode 1: RNN

```
# Initialize the RNN parameters (weights and biases)
# Define the activation function (e.g., sigmoid, tanh, ReLU)
# Define the forward propagation function for one time step
def rnn_forward(x, prev_h):
    # Calculate the hidden state (h) using the input (x) and the previous hidden state (prev_h)
    hidden_state = activation_function(Weight_input * x + Weight_hidden * prev_h + bias)
    return hidden_state
# Initialize the initial hidden state (h0)
# Loop over each time step in the input sequence
for t in range(num_time_steps):
    # Get the current input at time step t
    current_input = input_sequence[t]
    # Perform forward propagation to compute the current hidden state
    current_hidden_state = rnn_forward(current_input, prev_hidden_state)
    # Update the previous hidden state for the next time step
    prev_hidden_state = current_hidden_state

# Final output at the last time step
final_output = current_hidden_state

# Use the final output for further processing or prediction
```

3.8.2 LSTM

Long Short-Term Memory (LSTM) is made with both Long-Term Memory (LTM) and Short-Term Memory (STM) and uses the gate concept for simple and effective calculation . For a forget gate in an LSTM cell, forward pass equations are:

$$p_t = \sigma_g(W_p x_t + U_p h_{q-1} + b_p), \quad (2)$$

$$q_t = \sigma_g(W_q x_t + U_q h_{t-1} + b_q), \quad (3)$$

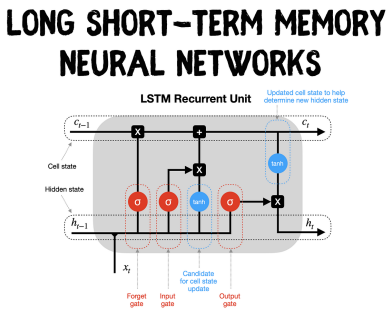
$$r_t = \sigma_g(W_r x_t + U_r h_{t-1} + b_r), \quad (4)$$

$$\tilde{s}_t = \sigma_s(W_s x_t + U_s h_{t-1} + b_s), \quad (5)$$

$$s_t = p_t \odot s_{t-1} + q_t \odot \tilde{s}_t, \quad (6)$$

$$h_t = r_t \odot \sigma_h(s_t). \quad (7)$$

Where x_t is the LSTM cell's input vector, p_t is forget gate's activation vector, q_t is the input/update gate's activation vector, r_t is the output gate's activation vector, h_t is the hidden vector, s _t is cell input's activation vector, s_t is state vector of the cell .(Fig. 3) shows an LSTM cell where the green rectangles are the layers, blue circle rectangle means the component-wise. Below to upper direction means the copy and upper to below direction means concatenate. In this figure, σ is the sigmoid function whose range is (0,1). Another activation function named tanh is also used here. That range is (-1,1). The tanh activation function is connected to the output end. So, its output is in the range of (-1,1).



Pseudo-code:

The pseudo-code of the LSTM model is :

Pseudocode 2: LSTM

Notations: In = Input, N = Number of neurons for this model, E = Repeat/Epochs

1. procedure LSTM (In, N, E)

Import library and datasets

2. Input Dataset with variable combinations.

Training LSTM

3. for In = 1 to End of input do

4. for N = 1 to 256 do

5. for E = 1 to 200 do

6. Add Softmax

7. Train LSTM

8. end for

9. end for

10. end for

11. return LSTM-metrics

12. end procedure

CHAPTER-4

IMPLEMENTATION

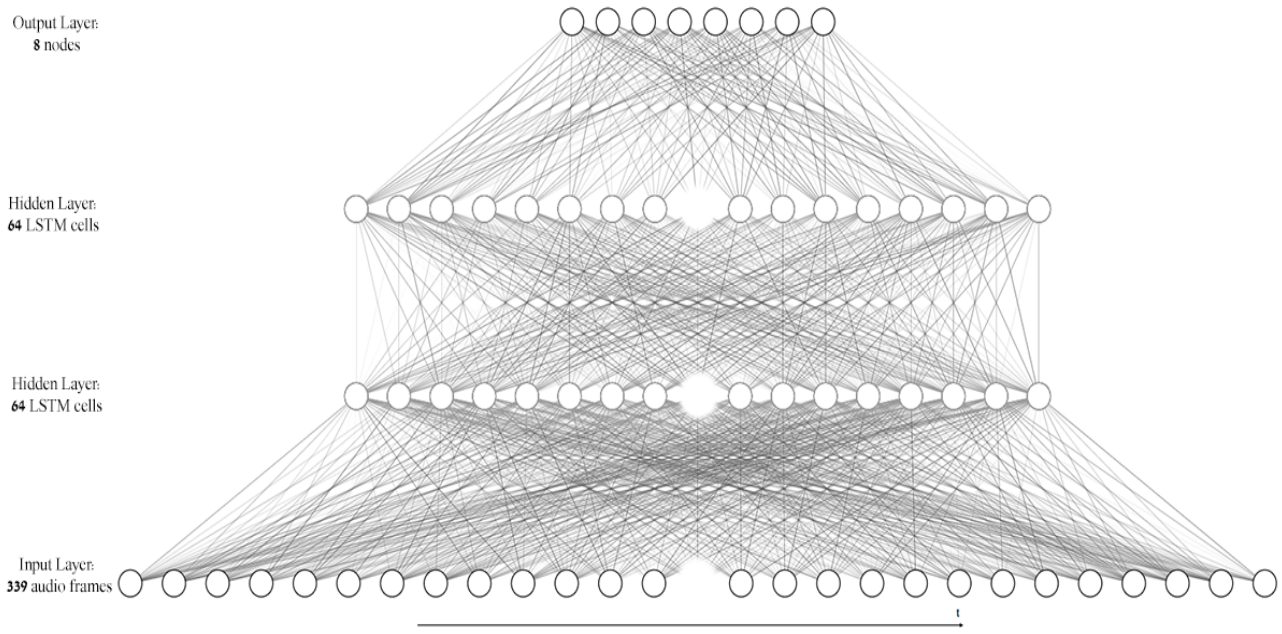
4.1 REQUIRED LIBRARIES AND PACKAGES:

```
import pandas as pd
import numpy as np
import os
import seaborn as sns
import matplotlib.pyplot as plt
import librosa
import librosa.display
from IPython.display import Audio
import warnings
warnings.filterwarnings('ignore')
conda update wrapt
!pip install tensorflow
import tenserflow
import keras
pip install tensorflow-gpu==1.14
!pip install keras
```

4.2 IMPLEMENTATION OF LSTM MODEL:

The model is executed with keras library, using 2 hidden LSTM layers with 64 nodes, and an output (dense) layer with 8 nodes, each for one emotion using the 'softmax' activation. The optimizer that led to the best results was 'RMSProp' with default parameters.

The batch size chosen is 23, which is a factor of all samples in the sets; train (3703), validation (368) and test (161).



4.3 FINAL DATA SETUP

In order to input the data into a model, a few adjustments should be made:

- The shapes of the features must be uniform, and in the 3D format:

(batch, timesteps, feature)

- Concatenating all features to a single 'X' variable.
- Adjustment of 'Y' with a 2D shape (keras library requirement)
- Split of X, Y to train, validation, and test sets.

- y_train and y_validation conversion to 'One-hot' vectors for classification purposes (y_test is being converted adjacent to the test)

```
from sklearn.model_selection import train_test_split
```

```
X_train,X_test,y_train,y_test= train_test_split(X,y, test_size=0.10, random_state=111)
```

```
X_train.shape, X_test.shape, y_train.shape, y_test.shape
```

```
((2520, 40, 1), (280, 40, 1), (2520, 7), (280, 7))
```

4.4 MODEL EVALUATION

The model has been evaluated using the following factors:

1. A visualization of the loss and categorical accuracy values trend during the train process.
2. A confusion matrix for visualizing the number of successful predictions of each emotion: for validation and test sets.
3. Model's prediction accuracy rates for each emotion: for validation and test sets.

```
from keras.models import Sequential
from keras.layers import Dense, LSTM, Dropout

model = Sequential([
    LSTM(123, return_sequences=False, input_shape=(40,1)),
    Dense(64, activation='relu'),
    Dropout(0.2),
    Dense(32, activation='relu'),
    Dropout(0.2),
    Dense(7, activation='softmax')
])

model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
model.summary()
```

OUTPUT:

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_1 (LSTM)               (None, 123)               61500

 dense_3 (Dense)             (None, 64)                7936

 dropout_2 (Dropout)         (None, 64)                0

 dense_4 (Dense)             (None, 32)                2080

 dropout_3 (Dropout)         (None, 32)                0

 dense_5 (Dense)             (None, 7)                 231

=================================================================
Total params: 71,747
Trainable params: 71,747
Non-trainable params: 0
_________________________________________________________________
```

4.5 TRAINING THE MODEL:

```
history = model.fit(X_train, y_train, validation_split=0.2, epochs=100, batch_size=512, shuffle=True)
```

OUTPUT:

```
Epoch 98/100
4/4 [==============================] - 2s 451ms/step - loss: 0.0024 - accuracy: 1.0000 - val_loss: 0.0375 - val_accuracy: 0.9901
Epoch 99/100
4/4 [==============================] - 2s 413ms/step - loss: 0.0020 - accuracy: 0.9995 - val_loss: 0.0384 - val_accuracy: 0.9901
Epoch 100/100
4/4 [==============================] - 2s 443ms/step - loss: 0.0025 - accuracy: 1.0000 - val_loss: 0.0479 - val_accuracy: 0.9901
```

CHAPTER-5

RESULT

5.1 ACCURACY

```
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
accuracy_score(y_test, y_pred_classes)
```

OUTPUT:

```
0.9892857142857143
```

Print the predicted emotion label

```
print("Predicted Emotion: ", predicted_emotion)
```

```
Predicted Emotion:  happy
```

CONFUSION MATRIX

confusion matrix

```
from sklearn.metrics import confusion_matrix

cm = confusion_matrix(y_test, y_pred)

print(cm)
```

```
[[41  0  0  0  0  0  0]
 [ 0 46  0  0  0  0  1]
 [ 0  0 42  0  0  0  0]
 [ 0  0  0 34  0  0  0]
 [ 0  0  0  0 40  0  0]
 [ 0  0  0  0  0 40  0]
 [ 0  0  0  0  0  0 36]]
```

It is a square matrix that summarizes the predictions made by the model against the true labels. [[TN, FP] [FN, TP]]

True Positive (TP): The predicted value matches the actual value. The actual value was positive and the model predicted a positive value

True Negative (TN): The predicted value matches the actual value. The actual value was negative and the model predicted a negative value

False Positive (FP): Type 1 error. The predicted value was falsely predicted. The actual value was negative but the model predicted a positive value Also known as the Type 1 error

False Negative (FN) : Type 2 error. The predicted value was falsely predicted.The actual value was positive but the model predicted a negative value. Also known as the Type 2 error

Accuracy: This is how we'll calculate the validation accuracy:

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN).$$

It represents the proportion of correct predictions.

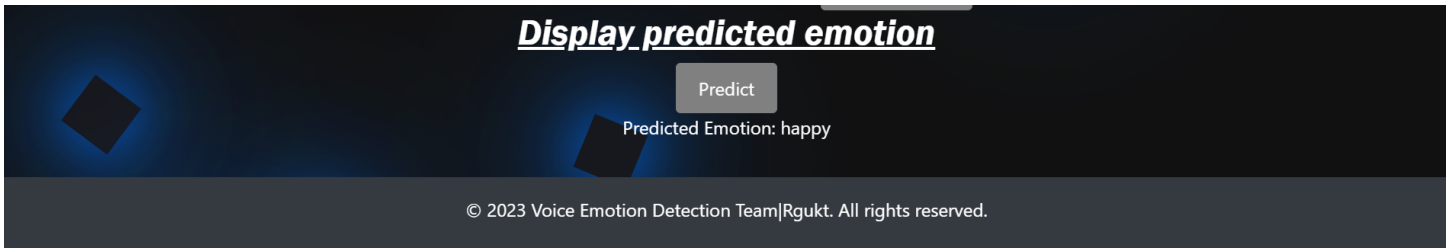
Precision: Also known as positive predictive value, it measures how many of the predicted positive instances were actually positive, calculated as TP / (TP + FP). Precision indicates the model's ability to avoid false positives.

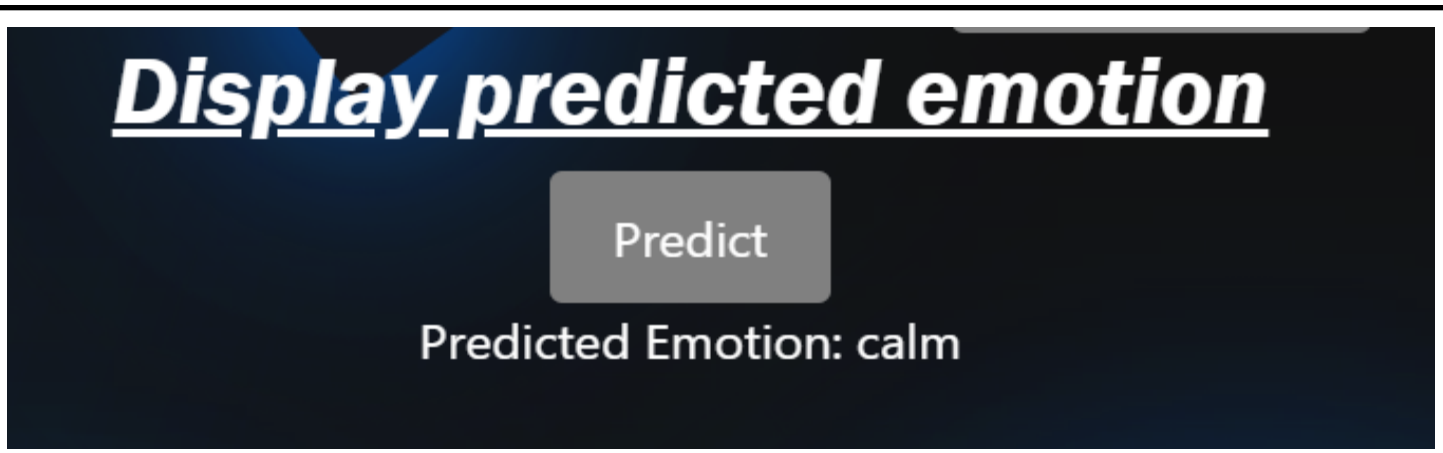
Recall: Also known as sensitivity or true positive rate, it measures how many of the actual positive instances were correctly predicted, calculated as TP / (TP + FN). Recall represents the model's ability to capture true positives.

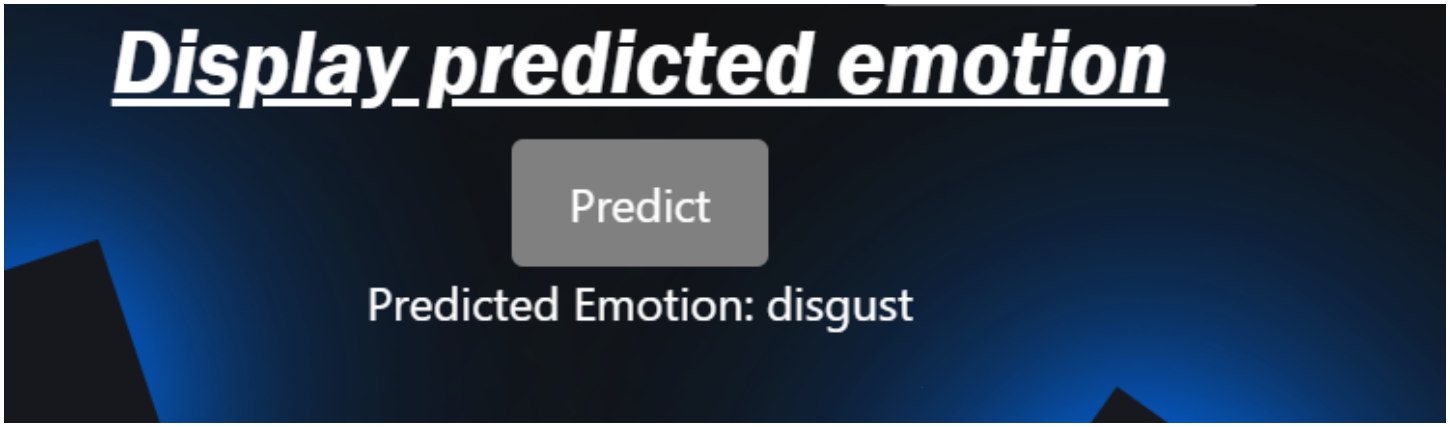
F1 Score: It is the harmonic mean of precision and recall, calculated as 2 * (Precision * Recall) / (Precision + Recall). The F1 score provides a balanced measure of precision and recall.

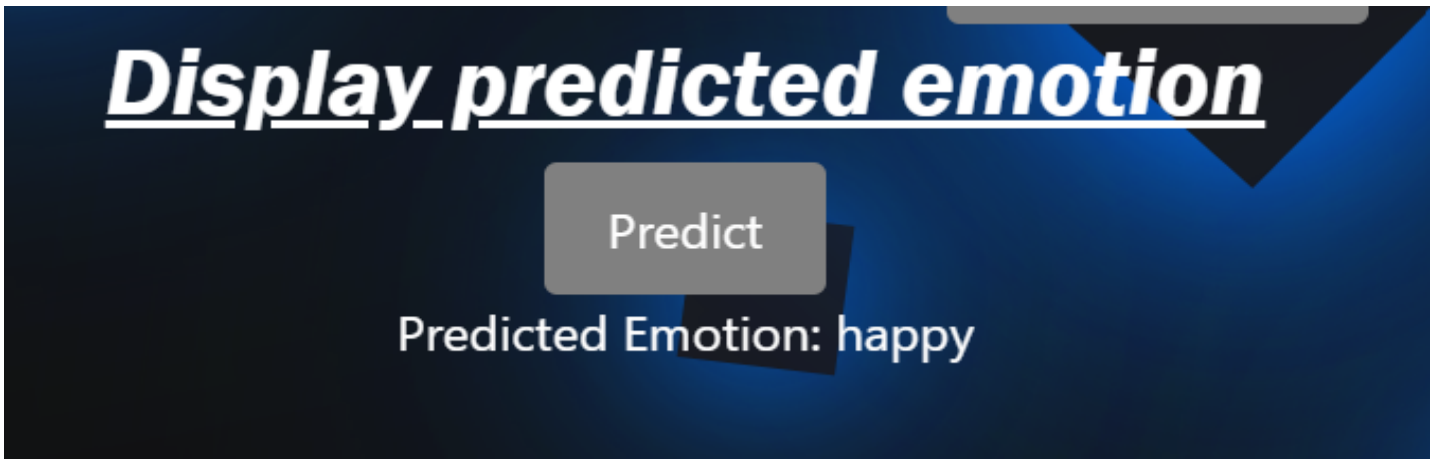
The confusion matrix helps you understand the performance of your voice emotion recognition model by revealing the distribution of correct and incorrect predictions across different emotion classes.

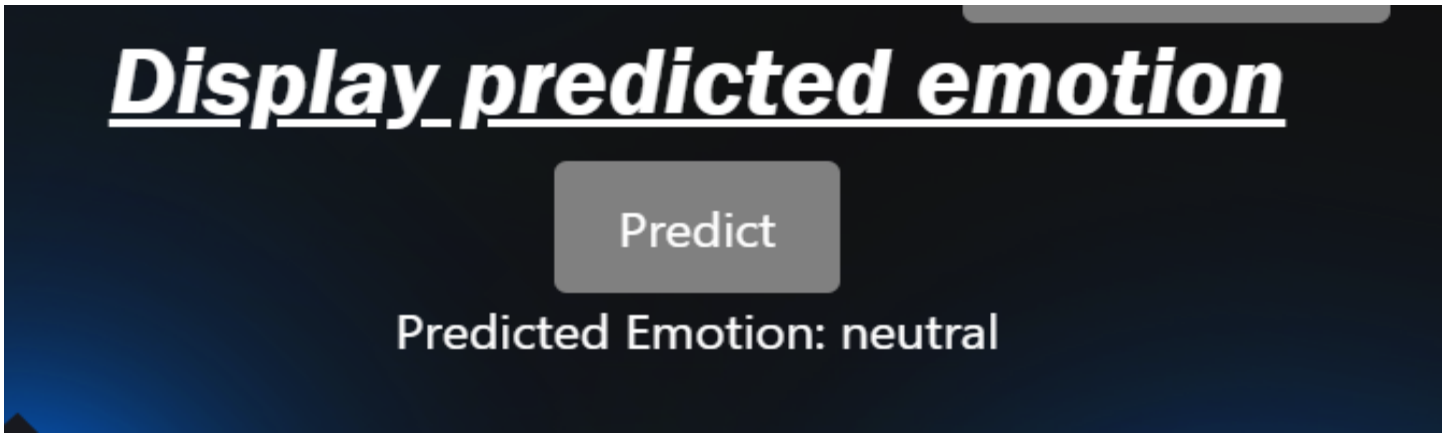
OUTPUT:

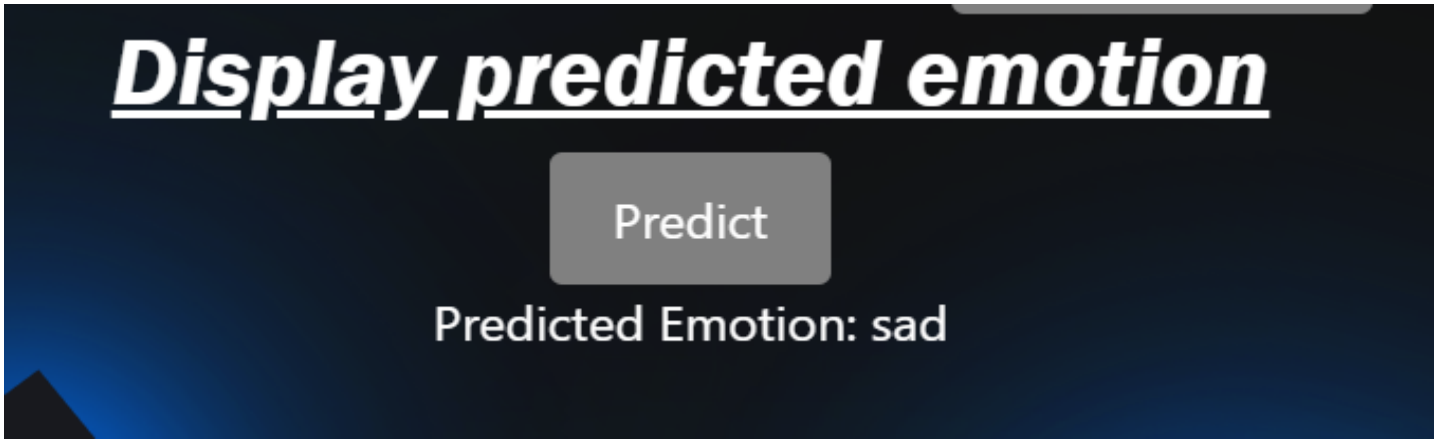



Display predicted emotion
Predict
Predicted Emotion: calm

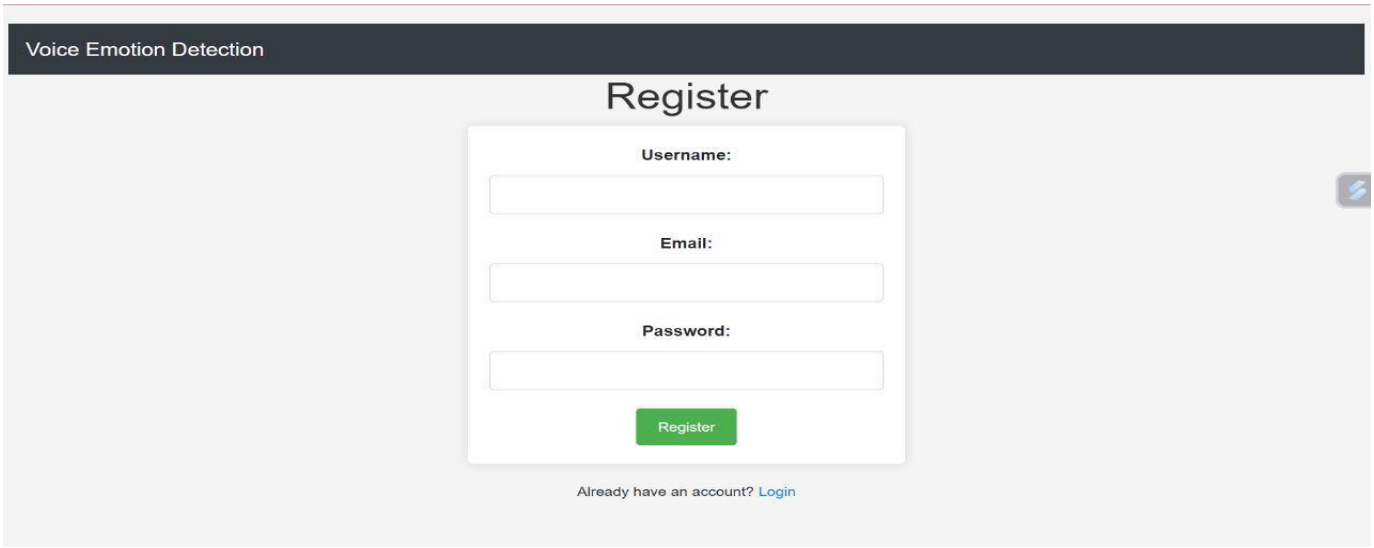

Display predicted emotion
Predict
Predicted Emotion: disgust


Display predicted emotion
Predict
Predicted Emotion: happy





Deployment (Webpage)



Contact

If you have any questions or feedback, please feel free to contact us using the form below:

Name:

Email:

Message:

Submit

About Voice Emotion Detection

Voice Emotion Detection is an innovative application that leverages machine learning and audio processing techniques to analyze and interpret human emotions from voice recordings. The system can accurately detect and classify various emotions such as happiness, sadness, anger, and more, based on the patterns and characteristics present in the recorded audio.

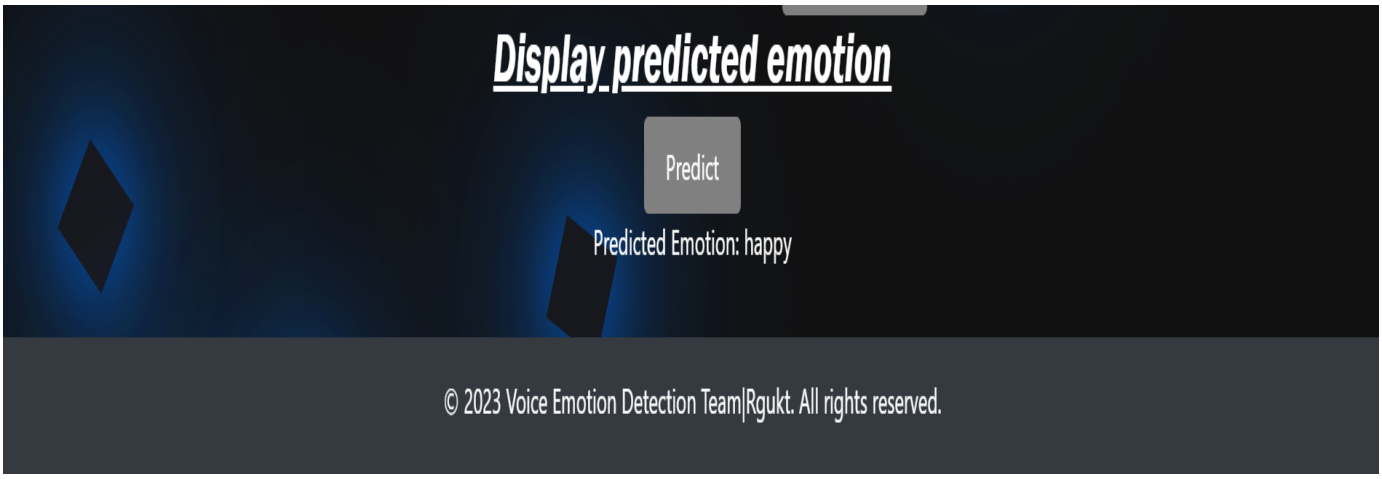
How it Works

The Voice Emotion Detection system utilizes a deep learning model trained on a large dataset of voice samples labeled with corresponding emotions. The model extracts relevant features from the audio signals, such as pitch, tone, and intensity, and applies advanced algorithms to classify the emotional state of the speaker.

Model Accuracy

After completing the ML model, we evaluated its performance using a test dataset. The model achieved an accuracy of 98%, indicating that it correctly classified 98% of the instances. This demonstrates the model's ability to make accurate predictions and suggests its effectiveness in solving the given task.. In future work, we aim to address these limitations by conducting additional evaluations on diverse datasets and performing rigorous testing in real-world settings. This will help further validate the model's accuracy and improve its applicability to a broader range of scenarios.

About the Team



CHAPTER-6

CONCLUSION

6.1 SUMMARY

We have achieved the main objective of the project, which is to identify emotion of a person using recurrent neural networks with Long short term memory. In order to meet this requirement we are using a dataset of 1440 files that include emotions like calm , happy ,sad ,fear ,disgust ,surprise and neutral. The system is tested under the LSTM machine learning model. And machine learning models usually accept numeric values as input so we convert our data to arrays before they are used in extraction of features . The feature used in this model is MFCC and is extracted using the librosa package. The extracted values are given as input to the developed LSTM model which uses these features and gives the final predicted emotion. The overall accuracy obtained in this model is 98 %. Furthermore, the accuracy of the model can be improved by clearing random silence from audio clips and adding more data volume by finding more annotated audio clips.

6.2 FUTURE SCOPE

Our future scope is to recognizing the patient's emotions using deep learning techniques. Automatically identifying the emotions can help build smart healthcare centers that can detect depression and stress among the patients in order to start the medication early. We focus on studying emotions' recognition from speech and audio-visual input and show the different techniques of deploying these algorithms in the real world. These two emotion recognition techniques can be used as a surveillance system in healthcare centers to monitor patients. We conclude the survey with a presentation of the challenges and the related future work to provide an insight into the applications of using emotion recognition.

6.3 REFERENCES

[1] Dsp.stackexchange.com. (2018). Framing an audio signal. [online] Available at: https://dsp.stackexchange.com/questions/27243/framing-an-audio-signal.

[2] Giannakopoulos, T. (2018). pyAudioAnalysis. [online] GitHub. Available at: https://github.com/tyiannak/pyAudioAnalysis.

[3] B. Liu, H. Qin, Y. Gong, W. Ge, M. Xia, and L. Shi, „„EERA-ASR: An energy- efficient reconfigurable architecture for automatic speech recognition with hybrid DNN and approximate computing,"" IEEE Access, vol. 6, pp. 52227– 52237, 2018

[4] N. Cummins, S. Amiriparian, G. Hagerer, A. Batliner, S. Steidl, and B. W. Schuller, „„Animage-based deep spectrum feature representation for the recognition of emotional speech,"" in Proc. 25th ACM MultimediaConf. (MM), 2017, pp. 478–484.

[5] T. Hussain, K. Muhammad, A. Ullah, Z. Cao, S. W. Baik, and V. H. C. de Albuquerque, „„Cloud-assisted multiview video summarization using CNN and bidirectional LSTM,"" IEEE Trans. Ind. Informat., vol. 16, no. 1, pp. 77–86,Jan. 2020