EMOTION RECOGNITION AND PROGRESSION TRACKING IN THERAPY SESSIONS USING MACHINE LEARNING

A project report submitted in partial fulfillment

of the requirements for the degree of

Bachelor of Technology

in

Electronics & Computer Engineering

by

THEERTHA KRISHNA	21BLC1175
AAKASH R. P.	21BLC1181
MADHAV JAY	21BLC1425

School of Electronics Engineering,

Vellore Institute of Technology, Chennai,

Vandalur-Kelambakkam Road,

Chennai - 600127, India.

November 2024

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Declaration

I hereby declare that the report titled ***Emotion Recognition and Progression Tracking in Therapy Sessions Using Machine Learning*** submitted by us to the School of Electronics Engineering, Vellore Institute of Technology, Chennai in partial fulfillment of the requirements for the award of **Bachelor of Technology** in **Electronics and Computer Engineering** is a bona-fide record of the work carried out by me under the supervision of ***Dr. Karthikeyan P. R.***.
I further declare that the work reported in this report, has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma of this institute or of any other institute or University.

Sign: _____

Name & Reg. No.: THEERTHA KRISHNA 21BLC1175

Date: 20.11.2024

Sign: _____

Name & Reg. No.: AAKASH R. P. 21BLC1181

Date: 20.11.2024

Sign: _____

Name & Reg. No.: MADHAV JAY 21BLC1425

Date: 20.11.2024

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

School of Electronics Engineering

Certificate

This is to certify that the project report titled ***Emotion Recognition and Progression Tracking in Therapy Sessions Using Machine Learning*** submitted by ***Theertha Krishna* (21BLC1175)**, ***Aakash R. P.* (21BLC1181)** and ***Madhav Jay* (21BLC1425)** to Vellore Institute of Technology Chennai, in partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology** in **Electronics and Computer Engineering** is a bona-fide work carried out under my supervision. The project report fulfills the requirements as per the regulations of this University and in my opinion meets the necessary standards for submission. The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma and the same is certified.

Supervisor

Signature: .....

Name: .....

Date:

Head of the Department

Signature: .....

Name: .....

Date:

Examiner

Signature: .....

Name: .....

Date:

(Seal of the School)

Abstract

The field of human-computer interaction concerns the fusion of emotion recognition from speech and its applications ranging from mental health care to customer service and multimedia analysis. The work conducted here deals with the design and evaluation of four machine learning models namely CNN, RCNN, LSTM, and Bi-LSTM using two datasets, namely CREMA-D and TESS, for emotion classification.

In the experiments, CREMA-D performed poorly (40%) and TESS was highly accurate (98-99%), primarily because of overfitting. To remedy the inconsistencies, the two datasets were combined to be balanced. The balanced merged dataset allowed the models to perform much better and consistently better across all of them with the highest achieved accuracy of LSTM.

The new application is also presented within the scope of the project: a web-based system adapted for therapists. They shall be able to upload files containing audio records of therapy sessions and analyze frequencies in emotions and emotion progression in time. This is the manifestation of the practical use of emotion recognition in speech for better quality mental health care services.

Comparison of the performances of the models, confusion matrices, accuracy and loss graphs, and the line graph pointing to how a change in emotion is presented over time. All these results provide insight into efficacy as well as robustness for the LSTM model in emotion detection across different datasets.

Acknowledgements

We wish to express our sincere thanks and deep sense of gratitude to our project guide, Dr. Karthikeyan P. R., Professor, School of Electronics Engineering, for her consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the project work.

We are extremely grateful to Dr. Ravishankar A, Dean Dr. Reena Monica, Associate Dean (Academics) & Dr. John Sahaya Rani Alex, Associate Dean (Research) of the School of Electronics Engineering, VIT Chennai, for extending the facilities of the School towards our project and for his unstinting support.

We express our thanks to our Head of the Department Dr. Annis Fathima A for her support throughout the course of this project.

We also take this opportunity to thank all the faculty of the School for their support and their wisdom imparted to us throughout the course.

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.

Contents

List of Figures

Chapter 1

Introduction

1.1 Background and Motivation

Speech is one of the most principal modes of human communication, characterized by a wide amount of emotional context that facilitates interpersonal interaction and decision-making. It further influences and has significant implications for such domains as mental health and customer service, in which understanding an emotional state is crucial.

For all the advancement in this field, challenges still arise and do not get surpassed. This includes speech variability, accents, and datasets. This project tries to link many datasets together and further compare their model performances to ascertain there is robust and reliable emotion recognition.

1.2 Project Statement

The project involves the design and development of an emotion recognition system using machine learning techniques. Four models, CNN, RCNN, LSTM, and Bi-LSTM, are used to test the performance of the approaches over three configurations of the dataset: CREMA-D, TESS, and a combined dataset.

Beyond mere model performance analysis, this paper devises a practical application of emotion recognition: a therapist-oriented web platform for emotion analysis and tracking over several therapy sessions.

1.3 Objectives

Improving the model performance using preprocessing and merging CREMA-D and TESS datasets.

Design and training of four models for speech emotion recognition tasks- CNN, RCNN, LSTM, Bi-LSTM.

Performance of the model are analyzed by evaluating the accuracy, confusion matrices, and graphical representation.

Cross-evaluation of performance is a model on individual and combined datasets.

Develop a web-based system for the therapist to analyze emotions with speech recordings.

1.4 Scope of the Project

The scope of this project includes:

Development of a machine learning pipeline for emotion detection based on speech. Combining datasets in order to avoid overfitting and variability issues Deployment of the best performing model, being the LSTM, in a web application. Practical evaluation of the utility of emotion recognition in improving mental health care services.

- Data Preprocessing: The CREMA-D and TESS datasets were merged and processed to an equal level.
- Model Training: Tests have been performed using CNN, RCNN, LSTM and Bi-LSTM models to confirm that which one was more suitable for the application of the model.
- Performance Evaluation: Calculation of confusion matrices, accuracy graph, and loss curve were considered in this all-around evaluative process.
- Web-based Application Development It is an application that therapists use to monitor and analyze trends of emotions in their sessions.

This project will show the feasibility as well as practical utility of machine learning for emotion recognition within real-world therapeutic contexts.

Chapter 2

Literature Survey

2.1 Introduction to Speech Emotion Recognition

Emotion recognition is one of the important areas in AI applications, gaining its applications from mental health care to adaptive human-computer interaction. Emotion recognition using speech is especially challenging because emotionally nuanced information can be drawn from an audio signal based on frequency, tone, and articulation. This review captures how work progressed into techniques about emotion recognition and highlights gaps that this study has addressed.

2.2 Machine Learning Techniques for Speech Analysis

Advancements in machine learning techniques have greatly contributed to speech signal processing and understanding in the last couple of years. Classical models involve KNN classification and Gaussian Mixture Models, which were used as a building block. Recent progress mainly relies on applying improvements carried out by deep learning methods to achieve higher accuracy.

2.3 Relevant Studies on Speech Emotion Recognition

This section discusses some of the most relevant studies within the field, focusing on the contribution and methodology presented:

2.3.1 M.R. Amer, B. Siddiquie, C. Richey, and A. Divakaran, "Emotion Detection in Speech Using Deep Networks"

The paper introduces the concept of deep neural networks for detecting emotions in speech. The paper states how deep networks pick hierarchical features from raw audio, important in giving out complex emotions. This foundational work supports the application of LSTM and Bi-LSTM in this project.

2.3.2 C. Prakash, Prof. V. B. Gaikwad, "Analysis of Emotion Recognition System through Speech Signal Using KNN GMM Classifier"

Here, the authors analysed the traditional machine learning classifiers such as KNN and GMM for emotion detection. Though they seem to be simple, their performance is not satisfactory with high-dimensional data, which calls for more complex techniques like RCNN and deep learning models studied in this paper.

2.3.3 F. Yu, E. Chang, YQ. Xu, HY. Shum, "Emotion Detection from Speech to Enrich Multimedia Content"

It talks about emotion detection that can be integrated into multimedia systems. This can be seen as being relatively practical to use in real life. Findings from this research actually motivated the idea of developing an application that is therapist-friendly.

2.3.4 D. Mamiev, A.B. Abdusalomov, A. Kutlimuratov, B. Muminov, T. Keun Whangbo, "Multimodal Emotion Detection via Attention-Based Fusion of Extracted Facial and Speech Features".

In this work, the authors propose a multimodal approach to emotion detection, fusing facial and speech features. The proposed mechanism of attention-based fusion highlights the need for the base of other strong feature extraction, a concept indirectly inferred in the deep learning model adopted by this project.

2.3.5 K. L. Ong, C. P. Lee, H. S. Lim, K. M. Lim, T. Mukaida, "SCQT-MaxViT: Speech Emotion Recognition With Constant-Q Transform and Multi-Axis Vision Transformer"

From last year, the paper is discussing an application of vision transformers for speech emotion recognition. Applying transformers basically shows them to be effective for encapsulation of temporal and spectral features; therefore, this research can be used as an inspiration for possible future developments on this paper.

2.3.6 Vijayanand G., Karthick S., Hari B., Jaikrishnan V. "Emotion Detection using Machine Learning"

This paper compares the various machine learning algorithms to be used for emotion detection and commented on the trade-offs between traditional and deep learning approaches. Their findings guided the selection of CNN, RCNN, LSTM, and Bi-LSTM for this project.

2.3.7 Gupta S., Kumar P., Tekchandani R.K., "Facial Emotion Recognition Based Real-Time Learner Engagement Detection System in Online Learning Context Using Deep Learning Models"

Although specifically focused on facial emotion recognition, the methodology of this paper highlights the use of real-time analysis, which is reflected in the web application developed for therapists.

2.3.8 Monisha G.S., Yogashree G.S., Baghyalaksmi R., Haritha P., "Enhanced Automatic Recognition of Human Emotions Using Machine Learning Techniques"

This paper focuses on advanced machine learning models for emotion recognition, emphasizing the importance of feature extraction and dataset balancing. The findings align with how this project seeks to combine the CREMA-D and TESS datasets to ensure balanced performance.

Chapter 3

Methodology

The overview of the datasets used, data preprocessing approaches, and machine learning models applied to the speech emotion recognition is therefore included in this chapter. As such, experimental setup and motivations for combined datasets and multiple models are integrated.

3.1 Proposed Architecture

The architecture system of this project is designed to handle audio data; meanwhile, emotion detection using the machine learning model is simultaneously followed by giving results through an internet-friendly web interface. The three core components of the architecture are Data Preprocessing, Machine Learning-based Emotion Detection, and Web Application Interface, ensuring modularity, scalability, and ease of maintenance. The proposed system consists of three main components:

1. **Audio Input and Preprocessing:** Raw audio files are processed to extract meaningful features for emotion recognition. Users upload audio files through the web interface, supported in common formats like .wav and .mp3.

2. **Machine Learning Models:** The extracted features are fed into three different models—Recurrent Neural Networks (RNN), Convolutional Neural Networks (CNN), Bidirectional Long Short-Term Memory (Bi-LSTM),RNN with LSTM—to compare their performance. Each model was evaluated during development, with Bi-LSTM selected for deployment due to its superior accuracy and ability to capture temporal dependencies in speech.
   The deployed Bi-LSTM model predicts one of the predefined emotional categories and returns confidence scores for each prediction.

3. **Web Interface Integration:** The best-performing model is deployed via a web application to visualize real-time emotion detection results.

The architecture of the system is built to provide a smooth and efficient pipeline for detecting emotions. Each component provides an essential role in the effectiveness of the system in terms of accuracy, user-friendliness, and scalability. Combining sophisticated machine learning techniques with practical usability provides a bridge between technology and therapy in mental health, thereby providing therapists with an effective tool to track and analyze emotions over time.

3.2 Dataset Description

This project uses two publicly available datasets for emotion recognition:

1. **CREMA-D (Crowd-Sourced Emotional Multimodal Actors Dataset)**

   Vocabulary with 7,442 audio files read by 91 actors in six emotional subsets: anger, disgust, fear, happy, neutral and sad. There are different accents and pronunciations that enhance variability but result in lower accuracy since speech patterns are complex in terms of complexities.

2. **TESS (Toronto Emotional Speech Set)**

   This consists of 2,800 recordings from two female speakers expressing seven emotions: anger, disgust, fear, happiness, pleasant surprise, sadness, and neutral. High accuracy because the inter-speaker variability is very low, which in general results in overfitting of models.

3. **Combination Dataset**

   Considering that both CREMA-D and TESS had some disadvantages for applications at an individual level, the two datasets were combined to provide a better representation. Therefore, a balanced set was developed that included a high degree of diversity in profiles as well as emotion categories. The generalization of models improved greatly in the merged datasets.

3.3 Pre-processing and Feature Extraction

Pre-processing steps ensure that the original audio data is compatible with the machine learning models:

1. **Audio Pre-processing:**

   Resampled all audio files to a consistent 16 kHz. Converted to mono channel to reduce computational complexity. Standardized file lengths using padding or trimming to 3 seconds.

2. **Feature Extraction:**

   - Mel-Frequency Cepstral Coefficients (MFCCs): Extracted 13 coefficients to capture spectral features relevant to emotion.
   - Delta and Delta-Delta Features: Captured temporal changes in audio for added context.
   - Spectrograms: Visualized frequency variations over time as inputs to CNN and RCNN.

3.4 Model Training

The following machine learning models were used for emotion classification: CNN, RCNN, LSTM, and Bi-LSTM. Each model was trained on the preprocessed spectrogram data, which was derived from audio files. The models were compiled using categorical cross-entropy loss to handle the multi-class classification problem and the Adam optimizer, known for its adaptive learning rates.

3.4.1 CNN Architecture

The CNN architecture was proposed to efficiently capture the spatial features of input spectrogram data. The proposed model contains three layers of 2D convolution followed by ReLU activation functions, each with an increasing number of filters. Max pooling along spatial dimensions reduces the spatial dimensionality, enabling the network to deal more efficiently with the input spectrograms.

- Input Layer : The input to this layer is the 2D spectrogram data reshaped to dimensions (40, 40, 1), which represents the 40x40 dimensions of the MFCC spectrogram and 1 for a single channel or grayscale.
- Convolutional Layers:

  1. Conv2D Layer 1: The layer has 64 filters with a size of (3, 3). The setting of padding is 'same', so spatial dimensions are preserved. ReLU is applied for non-linearity and dropout with rate 0.1 is used to prevent overfitting.

2. Conv2D Layer 2: Doubles the filters to 128, same (3, 3) size, 'same' padding, followed by a ReLU activation and dropout applied once more.
3. Conv2D Layer 3: The final layer of this block has 256 filters with the same size and padding. ReLU activation is used, as well as a dropout rate of 0.1 to keep the model in control.

- Pooling Layers:

1. MaxPooling2D Layer 1: It will down sample the spatial dimensions with a pool size of (2, 2) which would reduce the dimensions from (40, 40) to (20, 20).
2. MaxPooling2D Layer 2: The second application of the MaxPooling2D layer is done using the same pool size of (2, 2) and reduces the dimensions to (10, 10).
3. MaxPooling2D Layer 3: Final pooling layer with pool size (2, 2) in order to get the output to (5, 5) dimension in order to get the feature maps compact

- Flatten and Dense Layers:

1. Result is flattened into a one-dimensional vector after the convolutional and pooling layers.
2. Then follows dense layer with 128 units using ReLU activation in order for the feature mapping.
3. Adding a dropout layer with a rate of 0.5 just before the output layer for further regularization

4. The final dense layer should have the number of units equal to the number of emotion classes: len(emotion$_{l}abels$).$Activationshouldbesoftmaxinordertousemulti-classclassification$.

Strengths: CNNs are effective at extracting spatial features and patterns, making them suitable for analyzing spectrograms. Limitations: CNN struggles with learning temporal dependencies, which are crucial for speech data, as it only processes the input through convolutional layers. This architecture for CNN captures both the spatial features from the spectrograms and deeper hierarchical structures needed for accurate emotion classification. Application of ReLU activations, use of dropout to avoid overfitting, and the proper sizing of filters are ways of improving the model's ability to generalize to unseen data.

3.4.2 RCNN Architecture

The emotion recognition algorithm implements the RCNN architecture, adopting a combination of convolutional layers and a GRU layer to leverage spatial and temporal characteristics in spectrogram data. This hybrid model is effective as it captures both local patterns over the data and long-term dependencies, which can then come in handy for sequential emotion classification tasks.

1. Convolutional Layers: The RCNN architecture begins with two convolutional layers which are intended to extract spatial features from images representing spectrograms. These layers apply filters to the spectrograms for detecting patterns and local features. ReLU activation functions must be applied after every convolutional layer to introduce non-linearity to train complex patterns in the model. Max pooling layers must be applied after every convolutional layer so that spatial dimensions get reduced while the feature values get preserved.

2. GRU Layer: Following the convolutional layers, another pertinent part of the RCNN architecture is the GRU layer. GRU stands for Gated Recurrent Unit, which consists of a type of recurrent neural network layer that is capable of learning temporal dependencies in sequential data, such as the emotional progression in speech over time. In this case, the GRU layer processes the extracted sequence of features from the earlier convolutional layers, capturing the temporal relationships important to make an accurate classification of emotion. The GRU layer equips the model better to handle sequential audio data compared to just CNNs.

3. Dropout Layer: Dropout is applied after the GRU layer with an aim to prevent overfitting. In training, this technique of regularization simply randomly deactivates a fraction of the neurons, hence forcing the model to generalize better instead of depending on specific nodes.

4. Dense Layer: The model finally has a dense layer with softmax activation. This layer will output the probability distribution over the emotion class, allowing the model to predict the most likely emotional state based on the input features.

The RCNN architecture thus pays attention to spatial features as output by the convolutional layers as well as temporal dependencies through a GRU layer. This allows it to accommodate sequentiality more effectively in emotion-related speech data.

Limitation: The most significant limitation with the RCNN model is its higher computation complexity compared to the relatively simpler models like CNN. Its complexity,

coupled with the increased training time, especially when large data sets are involved, attributes to the addition of the GRU layer.

3.4.3 LSTM Architecture

The LSTM model was designed for the efficient capture of temporal dependencies within speech data for emotion recognition. LSTM networks are very suited to sequential data, such as audio signals, as it is possible to model long-range dependencies and remember information over time.

1. Dropout Layers: Dropout layers are placed following every LSTM layer to prevent overfitting. The dropout rate is 0.3, meaning randomly a fraction of neurons during the training process are dropped and forcing the model not to depend on certain nodes and to generalize better.

2. Batch Normalization: Batch Normalization is applied right after every LSTM layer in order to normalize the activations and gradients. That would speed up the process of training while improving the model's performance at reducing the magnitude of learning instability.

3. Dense Output Layer: This by itself creates a fully connected dense layer with softmax activation over for multi-class classification. The layer, therefore, brings up a probability distribution over the possible emotion classes, which would allow the model to predict the most likely emotional state given the learned features.

Since the LSTM model handles long-term dependencies very well in speech, the model gets better for sequential data, such as audio. Stacked layers of LSTMs allow learning complex patterns in temporal variables for the model with better accuracy in emotion classification.

Limitation: The primary limitation of the LSTM model is that convergence is slower than with simpler models, such as CNN; this is attributed to its complex network and hence an increased number of parameters. The training times may also be longer, especially for big datasets.

3.4.4 Bi-LSTM Architecture

Another architecture known as the Bi-LSTM model combines the strength of forward and backward LSTM processing towards richer temporal dependence in the input data.

Such an architecture is very useful where an application such as emotion recognition can think both past and future in a sequence context.

1. Bidirectional LSTM Layer. The model begins with a bidirectional LSTM layer which has 256 units. The layer will process the input sequence both ways, forward and backward, allowing the network to learn about the past as well as future temporal context in the audio signal. The argument return_sequences=True means that this layer needs to output the sequence of steps to further process.

2. Dropout Layers: Dropout layers with a dropout rate of 0.2 are used after the Bi-LSTM layer and after every successive layer. This prevents overfitting by randomly zeroing out a fraction of input units in training so that the model generalizes better to unseen data.

3. Second LSTM layer: After the Bi-LSTM layer, an LSTM layer 128 units was added. This LSTM layer was designed to process further sequence information but does not return sequences. It only feeds out the final state after processing the whole sequence.

4. Dense Layers The network has two fully connected layers with 128 and 64 units. These are followed by ReLU activation functions so the model can learn some non-linear relationships between learned features. Dropout layers of rate 0.2 come in the order after every dense layer, reducing overfitting.

5. Output layer: The model ends by applying a softmax output layer of 6 units, representing the 6 classes for emotions. This layer gives the probability distribution over the emotion classes and the most likely one in an input sequence.

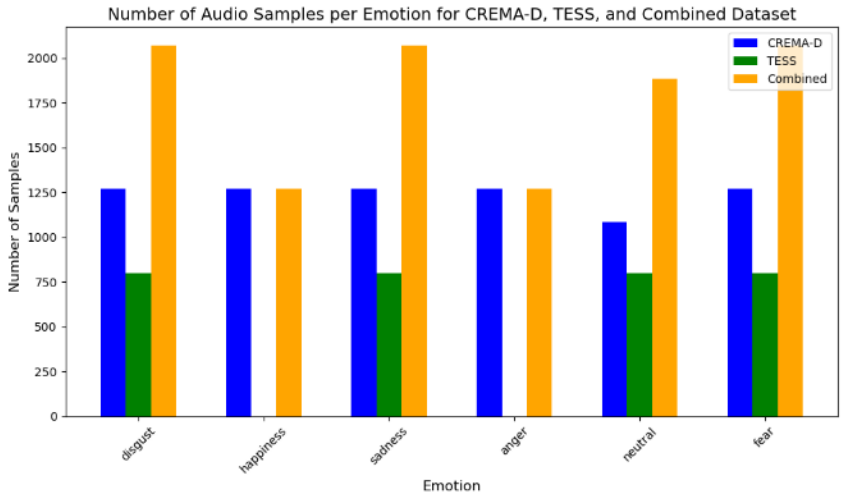

FIGURE 3.1: Number of audio samples per emotion for all datasets.

3.5 Experimental Setup

3.5.1 Model Training

- Frameworks and Libraries:
  - TensorFlow and Keras were used for model implementation.
  - Librosa was employed for audio preprocessing and feature extraction.
- Training Configuration:
  - Train-Test Split: An 80-20 split ensured ample data for both training and evaluation.
  - Optimizer: Adam optimizer with a learning rate of 0.001 for efficient gradient updates.
  - Loss Function: Categorical cross-entropy to handle multi-class emotion classification.
  - Spectrograms: Visualized frequency variations over time as inputs to CNN and RCNN.

3.5.2 Hyperparameter Tuning

Batch size: Tried values (16, 32, 64), and optimal trade-off for speed and stability was reached at 32 Epochs: Training is capped at 50 epochs along with early stopping based on the improvement in the validation loss. Dropout Rate: The rate in dense layers has been kept at 0.3 to avoid overfitting.

3.5.3 Training Process Optimization and Loss

- Loss Function: All the models are trained with categorical cross-entropy loss because it is a good match for multi-class classification problems.
- Optimizer: The Adam optimizer was used for the training, where the initial learning rate is taken to be 0.001. Adam takes into account sparse gradients along with even large datasets and gives efficient working.
- Early Stopping: To avoid overfitting, early stopping has been applied. It stopped the training if the validation loss had not improved in the last 10 epochs running consecutively.

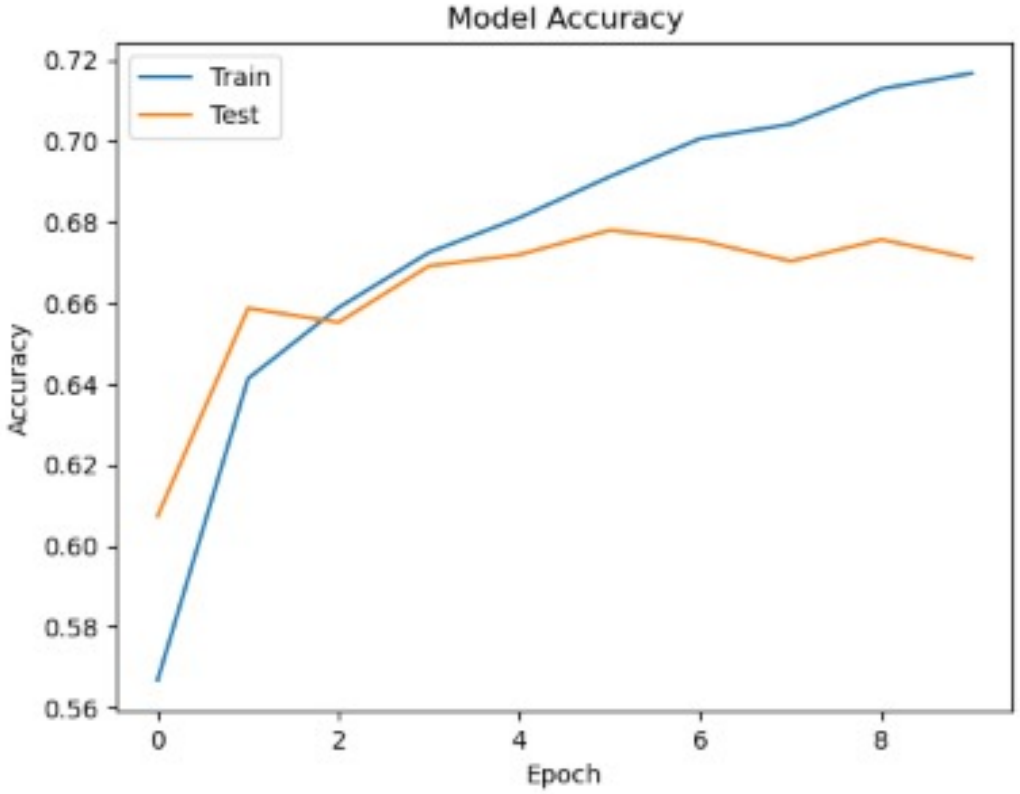

FIGURE 3.2: CNN Combined Model Accuracy

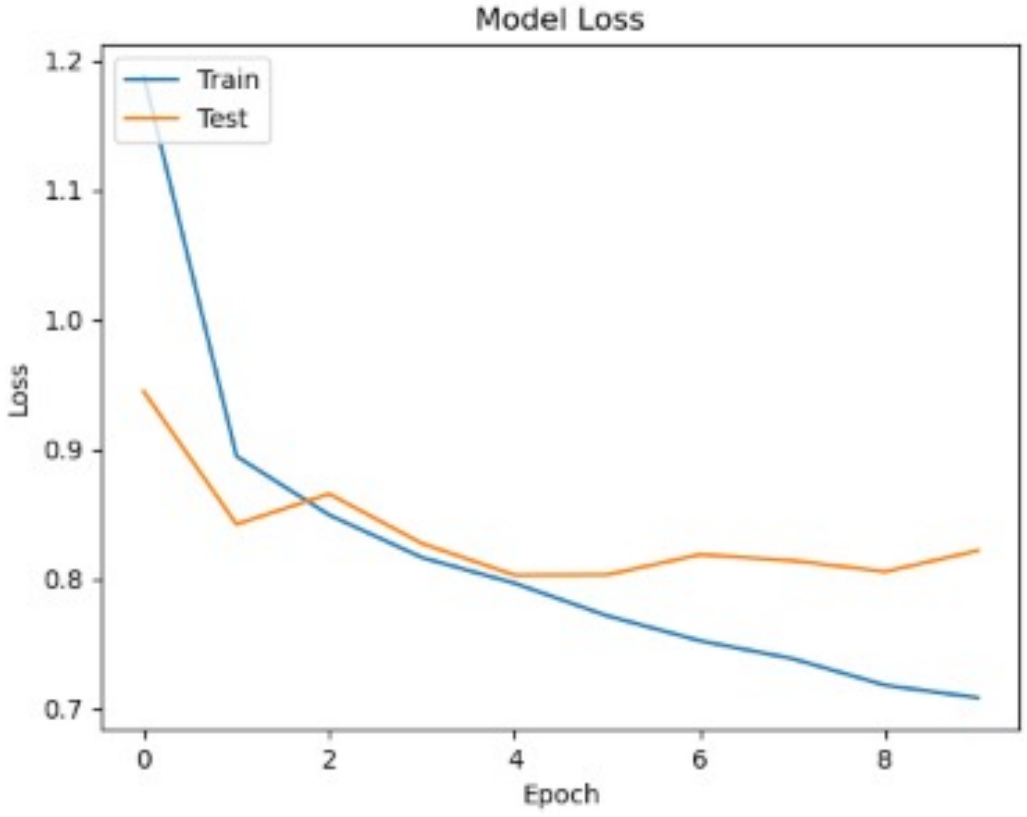

FIGURE 3.3: CNN Combined Model Loss

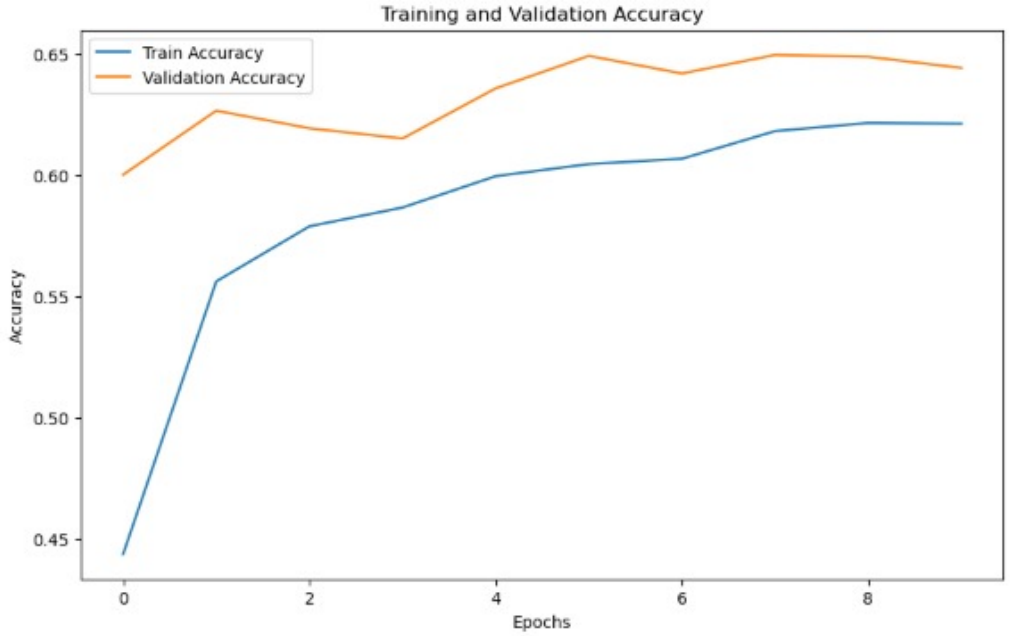

FIGURE 3.4: RCNN Combined Model Accuracy

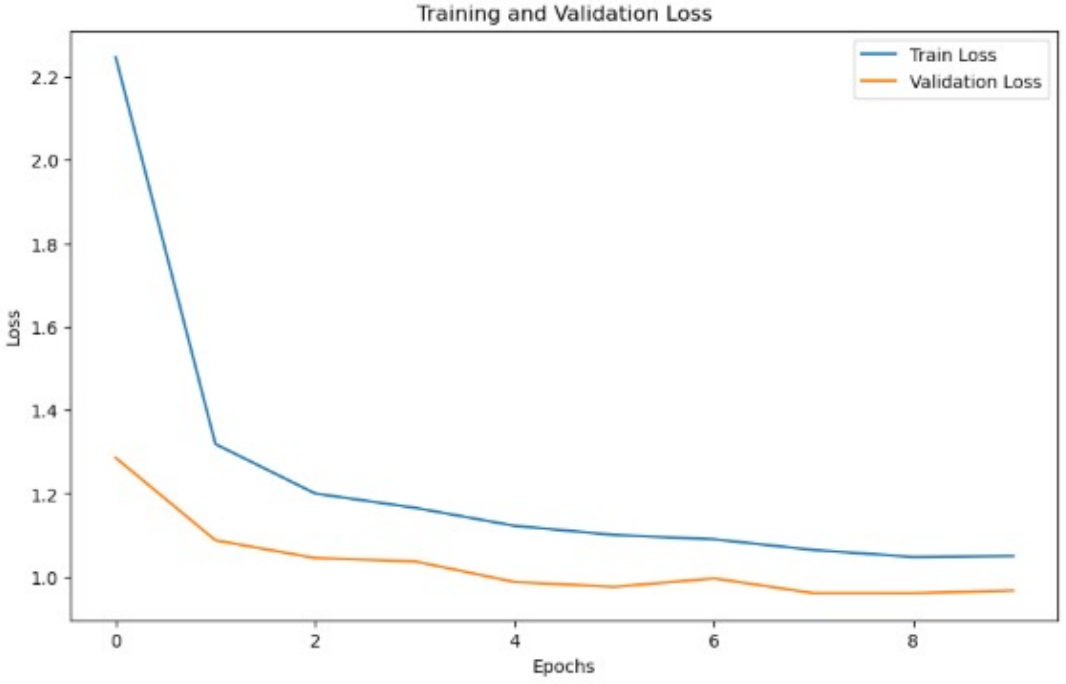

FIGURE 3.5: RCNN Combined Model Loss

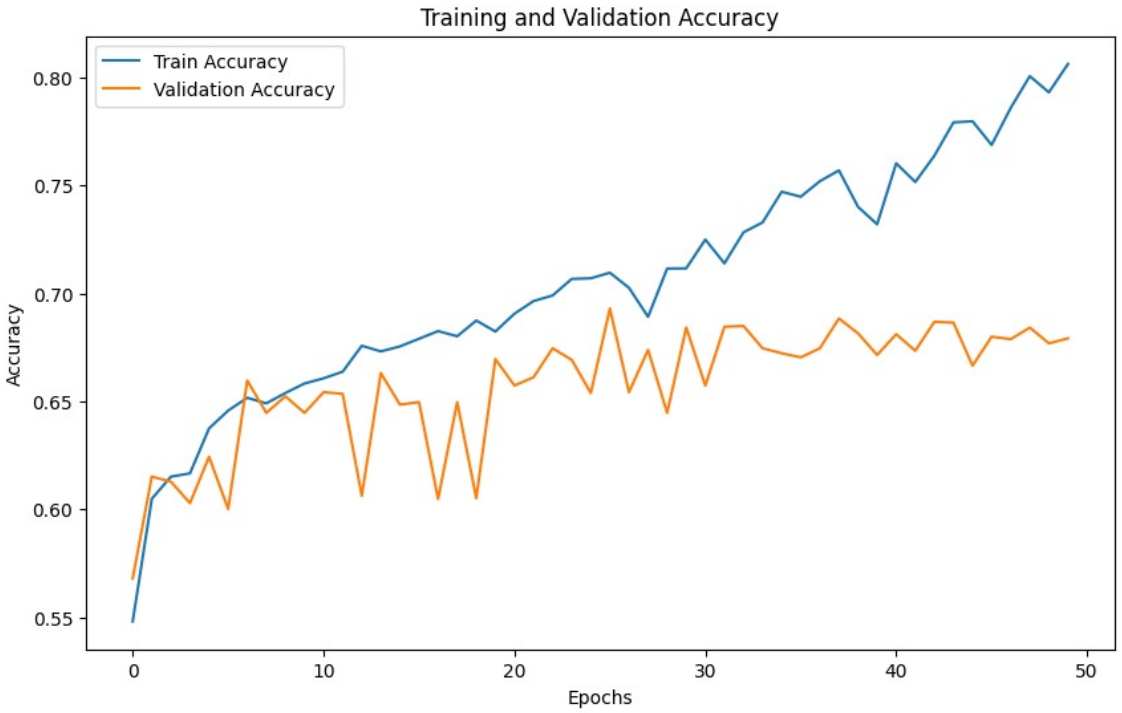

FIGURE 3.6: LSTM Combined Model Accuracy

FIGURE 3.7: LSTM Combined Model Loss

FIGURE 3.8: Bi-LSTM Combined Model Accuracy

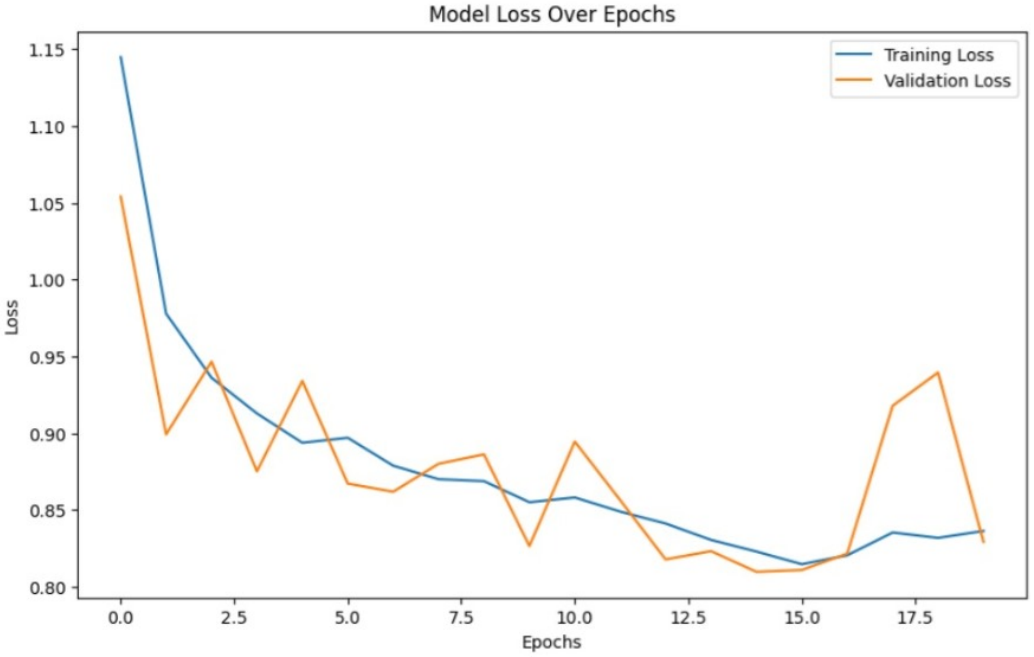

FIGURE 3.9: Bi-LSTM Combined Model Loss

3.5.4 Evaluation Metrics

To provide an in-depth assessment of the performance of emotion classification models, multiple metrics were considered to determine overall accuracy as well as class-wise performance. The suite of evaluation metrics used gives insights into how well the models perform in different aspects of classification, especially in highly imbalanced datasets or datasets with multiple classes. Such evaluation metrics include:

1. Accuracy

   Definition: Accuracy is the number of correct predictions, both true positives and true negatives, compared with the total number of predictions made. It is one of the most straightforward methods used to measure the performance of the model.

   Formula:

   Accuracy = $\frac{NumberofCorrectPredictions}{TotalNumberofPredictions}$

   Usefulness: Accuracy gives a general idea about how good the model will predict samples within the classifier. However, in a class-imbalanced dataset where one class is significantly dominant, accuracy becomes misleading as a model can predict the majority class for all samples and achieve high accuracy as well.

2. Precision

   Definition: Precision, alternatively called positive predictive value, measures the percentage of accurately predicted positive instances, or true positives, of all instances classified positive, that is, true positives plus false positives. Formula:

   Precision = $\frac{TruePositives}{TruePositives+FalsePositives}$ Utility: Precision is very useful in cases where high costs are associated with false positives. To illustrate, in emotions classification it would mean a bad wrong classification of an emotion in therapy or customer service applications

3. Recall (Sensitivity)

   Definition: Recall, or sensitivity, is the ratio of the number of true positive instances predicted by the model to all actual positive instances that exist (true positives + false negatives).

   Formula:

   Recall = $\frac{TruePositives}{TruePositives+FalseNegatives}$ Usefulness: Recall is crucial for applications in which the capture of every possible instance of a class is more critical (such as the detection of all instances of a given emotion).

4. F1-Score

   Definition: F1 score is the harmonic mean of precision and recall. It gives a balanced measure that accounts for both false positives and negatives. This makes

it very applicable when dealing with highly imbalanced datasets.

Formula:

F1-Score = $2 * \frac{PrecisionxRecall}{Precision+Recall}$ Usefulness: The F1-score is valuable in cases where a balance between precision and recall is needed. In emotion recognition, it is beneficial to avoid false positives (high precision) and yet capture all the emotions that should have been detected (high recall).

5. Confusion Matrix

   Definition: The confusion matrix is a tabular view of the true vs. predicted class labels of a classification problem. It breaks down the model's performance on classes. Usefulness: The confusion matrix proves helpful in knowing exactly where the model is getting things wrong, say, classifying one emotion as another. It becomes particularly useful when it points out possible imbalances in classes and misclassifications between similar emotions, like "neutral" with "happy," which share many features in the audio.

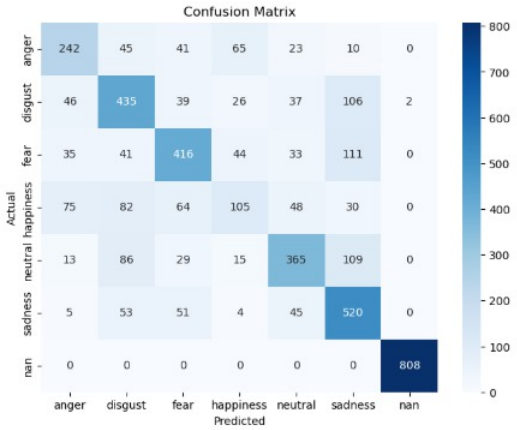

FIGURE 3.10: CNN Confusion Matrix

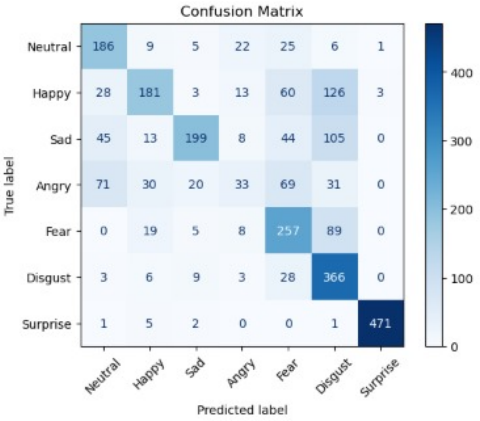

FIGURE 3.11: RCNN Confusion Matrix

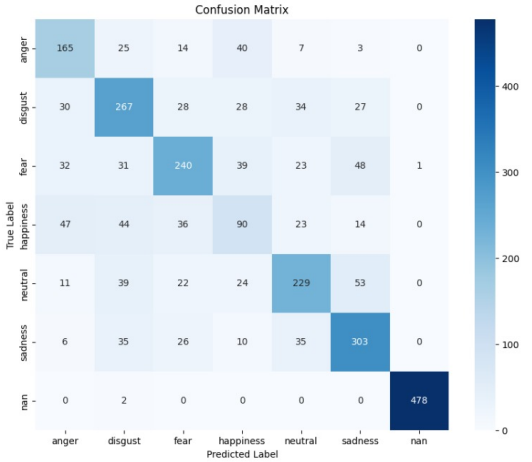

FIGURE 3.12: LSTM Confusion Matrix

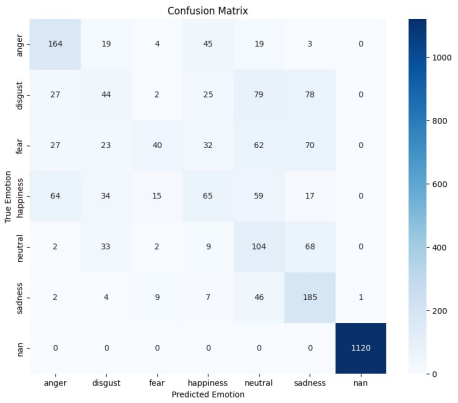

FIGURE 3.13: Bi-LSTM Confusion Matrix

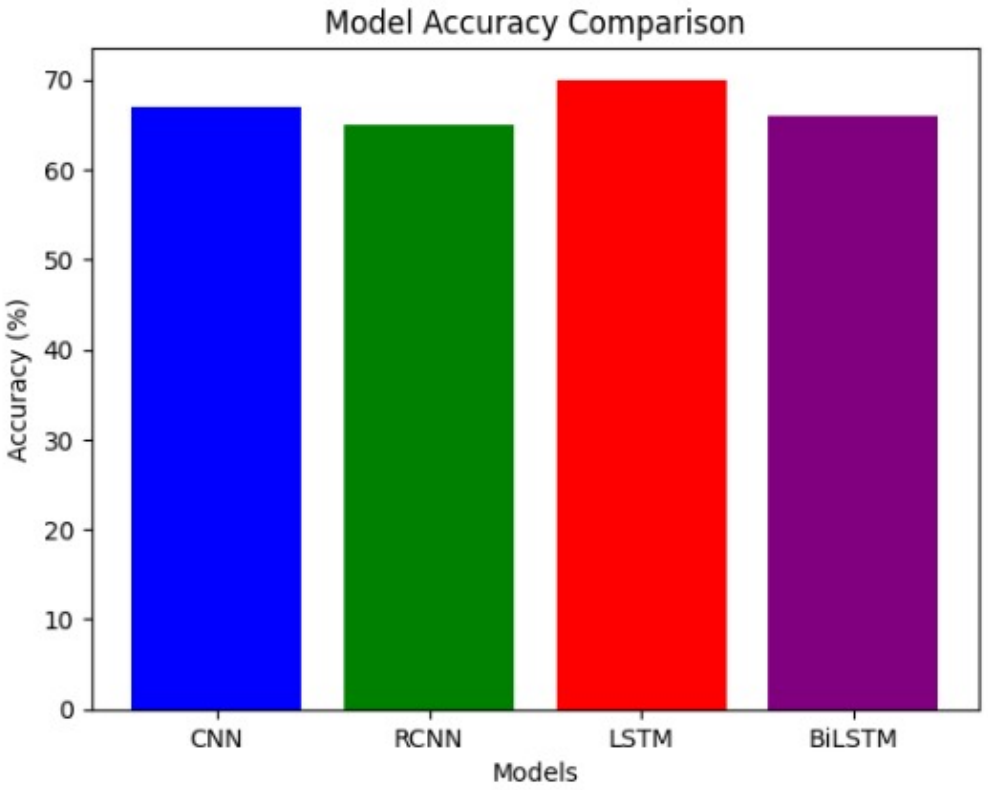

FIGURE 3.14: Bar Graph Comparing Accuracies

3.6 Dataset Comparisons

The models performed differently on different datasets and across different emotions:

1. CNN: Primarily excelled in high-energy emotional expression categories, such as anger and happiness, but fared worse with less high-energy emotions, including sadness and disgust.

2. RCNN: Represented major improvement over CNN in the recognition of emotions with well-defined temporal features like surprise and disgust, though sometimes confusing between anger and fear emotions.

3. LSTM: It was the best of all models because it is able to capture long-range dependencies in speech and, indeed, was the most consistent across all emotion categories.

4. Bi-LSTM: Underperformed by some margin because it demanded a greater computational overhead; however, it made sound progress at detecting transitions between emotions.

Chapter 4

Implementation

4.1 Introduction

This chapter implements a web application for speech emotion detection where the code would consist of recognition utilizing machine learning models. Users upload an audio file that will then process to predict the state of the emotion of the speaker. It is built using the Flask lightweight Python web framework with the use of a pre-trained machine learning model for emotion classification. This project brings together audio feature extraction using Mel-frequency cepstral coefficients (MFCCs) with deep learning models so as to predict emotions ranging from anger, happiness, and sadness among others in the speech.

4.2 System Architecture

In summary, the system consists of multiple components that work in harmony with one another: Frontend: User interface is developed using HTML, CSS, and JavaScript. Interactive elements are buttons, forms, dynamic content display, etc. Backend: It will use Flask, the Python web framework for developing the backend. Flask will be used to route HTTP requests, for serving static content, user authentications, and invoke the machine learning model for predicting the emotion. A neural network developed using Keras is deployed to classify the emotions by a trained model of a machine learning algorithm for processing the audio data. Session Management: Application provides secure access to the user by session management in which users are to log in before accessing any emotion detection service.

4.2.1 System Workflow

1. **User Authentication:**

   The user logs in to the system on a login page login.html; this restricts access to the functionality of emotion detection in the system. The credentials are checked against a primitive dictionary of usernames and passwords. After authentication, the user is navigated to the main page index.html : here, the different segments of the application come up for the use of the user, viz. profile, emotion prediction, and session history.

2. **Emotion Prediction:** User can download and upload a .wav audio file for the "Predict Emotion" section. The file is received on the backend, saved to a temporary folder, and then passed to a function to extract MFCC from an audio file by using the librosa library. The features are passed to the pre-trained machine learning model to predict emotion in the speech. The returned prediction is accompanied by the probability of each emotion. The result comes with a chart on the web interface where the different emotions and their probabilities are viewed.

3. **Session History:** There is also, for each emotion prediction, its probability and timestamp in the session history. These allow users to track the outcomes of their previous emotion detection. The session history can be viewed in a distinct section. Thus, users could easily see a table of the session, that is, the emotion predicted together with the related probabilities. Users are at liberty to clear the history too.

4. **Logout:** Users may also log out from the system at any time, which leaves their session and directs them to the login page.

4.3 Back-End Implementation

The backend setup has been done with Flask, a lightweight web framework where it is extremely easy to manage routes, requests, and templates. The main operations of the backend are outlined as follows:

(a) Flask Routes:

- Login Page: A page where users enter their credentials.
- Authentication: Authenticate users by checking their credentials against a predefined dictionary. If the process is successful, a redirection to the index page occurs.

- Index: The main page after the user is authenticated, which it offers different pages (profile, emotion prediction, and history).
- Predict: This deals with the POST request from the front end. It processes the uploaded audio file, extracts MFCC features, uses the trained model, and predicts the emotion. It then returns the outcome as a JSON response .
- Logout: This logs out the user and deletes his/her session

(b) Model Integration: We load the machine learning model from a file with the extension .keras using the function Keras load_model. The model is trained on datasets like CREMA-D and TESS. The machine learning model predicts one of the seven emotions such as anger, disgust, fear, happiness, neutral, sadness, or surprise.

- Audio Processing: The extract_mfcc function loads an audio file using the librosa library, extracts the MFCC features, and readies those for entry into the model.
- Prediction: The processed features feed through the model where predictions can be made. The model returns a probability distribution for each emotion, and the emotion with the highest probability is selected as the outputted emotion

(c) Session Management: Flask's session object takes care of the login status of the user. This means only authenticated users are able to make use of the emotion detection functionality. When a user logs in, the application will maintain that session of the user, and their name will display on the main page. The session continues till the logout process.

(d) File Handling: The application saves the uploaded audio file temporarily before processing. Once it makes its prediction, it deletes the file to free up that space.

4.4 Front-end Implementation

The frontend is clean and user-friendly in design. Some of the features included therein are the following:

(a) Login Page (login.html)
This page is where a user logs in to the system using a username and password. If the login credentials are wrong, then a message is displayed. Basic HTML and CSS is used for styling the page layout

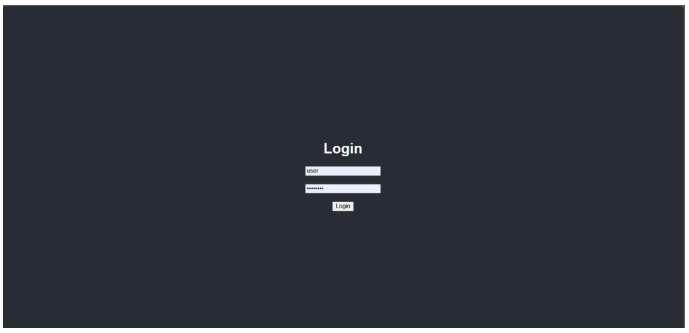

FIGURE 4.1: Login Page

(b) Main Page (index.html)

Sections present in this page include Profiling, Emotion Prediction, and Session History.

- Sidebar: This uses navigation links to switch sections. The user can view his or her own profile, make predictions, or view session history.

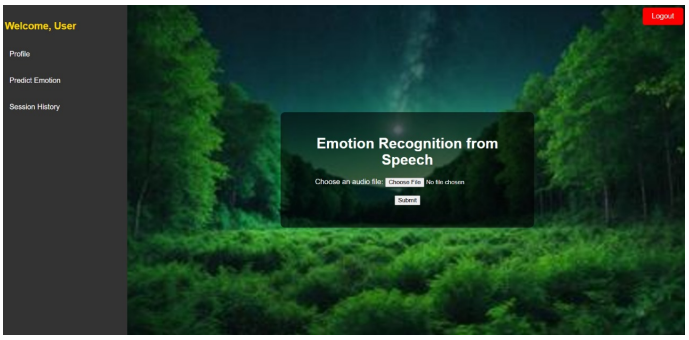

FIGURE 4.2: Home Page

- Emotion Prediction: A form is presented, which has the facility to upload audio and submit it for emotion prediction. Results are displayed dynamically, which includes a bar chart of the probability distribution of emotions.

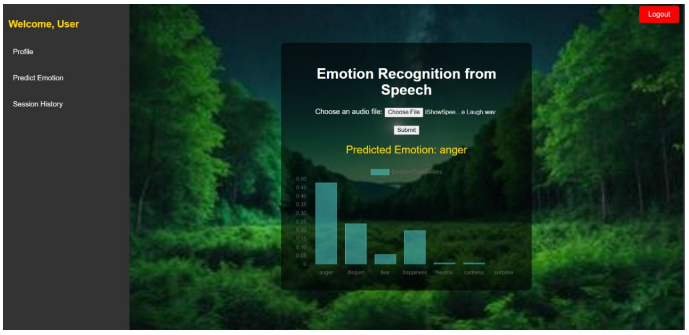

FIGURE 4.3: Emotion Prediction

- Session History: This group of screenshots depicts a table of previous emotion predictions, the related probabilities for each, and an option to clear history.

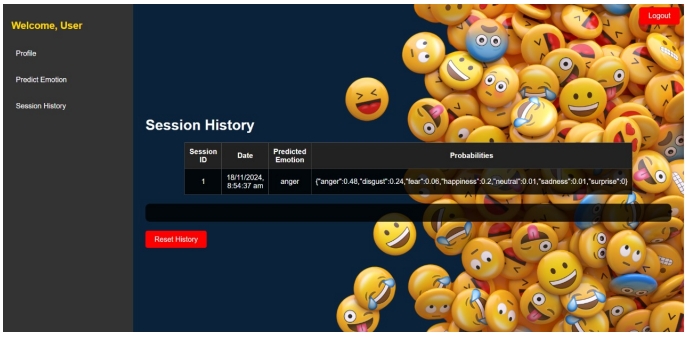

FIGURE 4.4: NSession History

(c) JavaScript Features:

The form will be submitted asynchronously. The file is submitted using a call to the Fetch API to send to the server. Then, upon receiving the response, it updates the page with the prediction result and a chart from Chart.js that plots the probabilities of the emotions

Chapter 5

Results and Discussions

This chapter presents the outcomes of the experiments, including model performance metrics, visualizations, and comparative analyses across datasets. It explores the implications of the results, which might lead to interesting avenues and practical utility in the developed system.

5.1 Performance on Individual Datasets

5.1.1 CREMA-D Dataset

- Results: All the models performed terribly on this dataset with accuracies ranging from 40-50%. The accents, speech patterns, and pronunciations varied significantly and were a challenge to consistently classify the emotions. LSTM and Bi-LSTM performed a little better than CNN and RCNN since they can extract the temporal dependencies within the sequential data.
- Challenges: Acoustic features that overlap between emotions such as fear and sadness easily misclassified emotions because of overlapping acoustic features. Diversity is a strength and a challenge for the given dataset since it is similar to real-world scenarios but calls for more sophisticated models for high performance.
- Key Insight: CREMA-D: The dataset calls for models that can work with this kind of variability in multiple diverse datasets.

5.1.2 TESS Dataset

- Results: All models attained extremely high accuracy - 98%, 99%. Only two profiles of speakers-low variability facilitated the learning process, causing overfitting.

- Observations: CNN and RCNN worked almost like LSTM and Bi-LSTM since the dataset was not complex. However, models failed to generalize and thus didn't perform as well when they were tested with unseen and diverse data.

5.2 Results on Combined Dataset

The dataset resolved the problems associated with CREMA-D-high variability and low accuracy-and TESS-low variability and overfitting.

- Overall Accuracy:

  LSTM performed the best, with an 85% level of accuracy. Bi-LSTM came second, at 83%. CNN and RCNN performed in the midrange, achieving accuracies of 75LSTM and Bi-LSTM performed a little better than CNN and RCNN since they can extract the temporal dependencies within the sequential data.

- Key Improvements:

  A balanced representation of emotions is portrayed across diversified speaker profiles by the combined dataset and improves the generalization ability of the models. Temporal models like LSTM and Bi-LSTM captured emotional nuances over time, enhancing classification accuracy.

Model	Accuracy	Precision	Recall	F1 Score
CNN	0.67	0.65	0.66	0.65
RCNN	0.65	0.67	0.65	0.63
LSTM	0.70	0.71	0.69	0.72
Bi-LSTM	0.66	0.66	0.67	0.69

Table 5.1: Performance Metrics Table

5.3 Visualization of Results

5.3.1 Confusion Matrices

Confusion matrices were generated for all models across datasets to evaluate classification performance for each emotion. These matrices provided insights into:
Class-wise Accuracy: Highlighting which emotions were well-classified and which were commonly misclassified.
Patterns of Misclassification: For example, emotions like fear and sadness were often confused due to their overlapping acoustic features.

5.3.1.1 CREMA-D:

- Observations:

  High misclassification rates for fear and sadness. Neutral and happiness were better distinguished by the clear pitch and energy.

- Insights:

  The difference in accents and speech patterns between individuals in CREMA-D was problematic for models to learn to generalize .

5.3.1.2 TESS

- Observations:

  All emotions classified near to a hundred percent.

  But the performance was due to overfitting rather than good generalization

5.3.1.3 Combined Dataset

- Observations:

  Fear and sadness are classified much better.

  Spatial models (CNN, RCNN) couldn't keep up with Temporal models such as LSTM and Bi-LSTM, which exploited sequential patterns in speech.

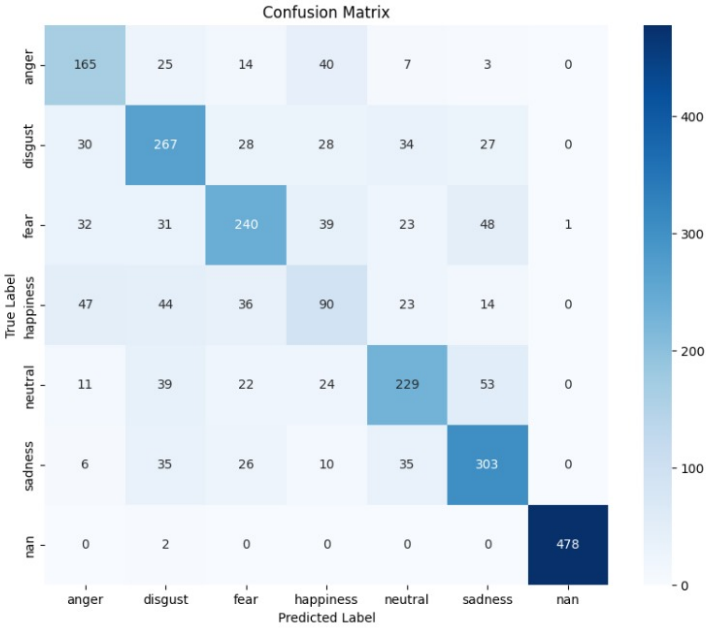

FIGURE 5.1: Confusion Matrix of LSTM Model

5.3.2 Model Accuracy and Loss Curves

The training and validation accuracy and loss curves were highly enlightening regarding model learning and generalization:

5.3.2.1 CREMA-D:

The curves had slow convergence, showing the complexity of the dataset used.

Temporal models were more stable in validation accuracy than CNN and RCNN.

5.3.2.2 TESS

Fast convergence with early plateau for validation accuracy.

Overfitting occurred since the training loss was continuously going down while validation loss never increased or went slightly upwards.

5.3.2.3 Combined Dataset

Smoothe learning curves with less overfitting both for LSTM and Bi-LSTM.

CNN and RCNN overfitted a little because these have less ability to cope with sequential dependency.

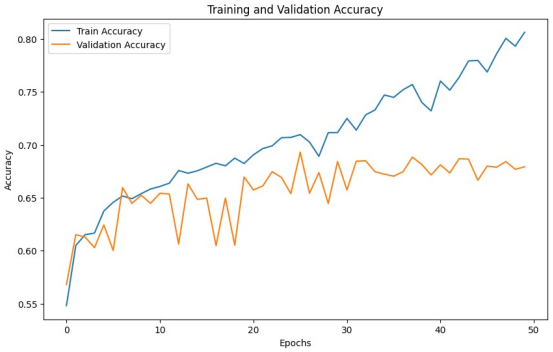

FIGURE 5.2: Model Accuracy of LSTM

FIGURE 5.3: Model Loss of LSTM

5.3.3 Emotion Evolution with Time

For longer audio files, line graphs were used to present the temporal evolution of emotion. For this, the graphs clearly depicted:

- Dynamic Evolution of Emotion: It may be that one speaker's emotion changes from neutral to happiness and then turns into sadness.

- Therapeutic Insights: Therapists could use these graphs to analyze emotional shifts within sessions, helping to assess progress or identify critical emotional moments.

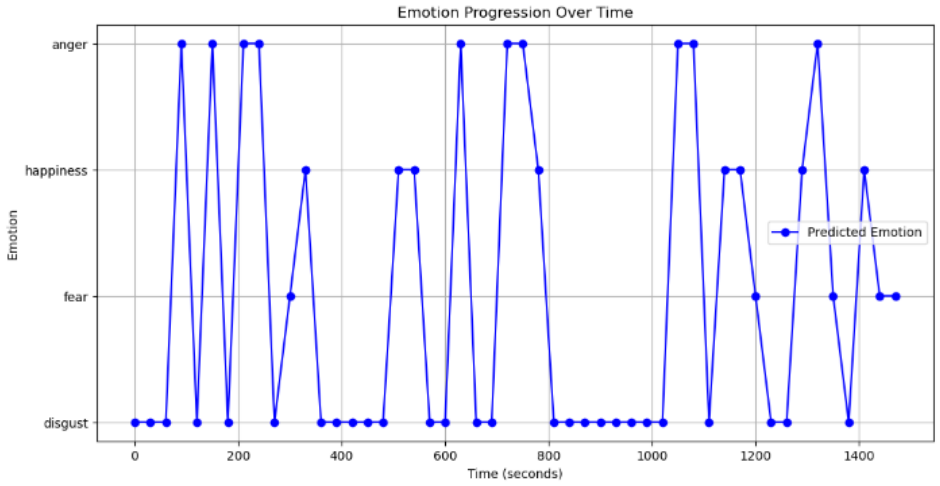

FIGURE 5.4: Emotion Progression over Time

5.4 Website Implementation

Building upon the results from this study, we have applied it into a website where therapists upload audio recordings of the therapy sessions, and the system analyzes the frequency and evolution of emotions.

- Features: The website allows users to upload audio files, and the system processes those files for real-time emotion detection. The output includes:
  - Frequency of occurrence of each detected emotion
  - Emotion change over time on a line graph
- Therapist Usage: The application allows a therapist to observe the emotional state of the patient from session to session and obtain vital information regarding his or her emotional development and underlying psychological issues.

5.5 Limitations and Future Work

Despite its effectiveness, the LSTM model is not thoroughly optimized yet. It gets confused easily with less intense emotions. This can be alleviated either through fine-tuning or with other additional features like prosody and pitch. Other advanced architectures, like transformers or attention mechanisms, may also help boost model performance.

5.6 Possible Applications

As such, this emotional detection system has serious potential in therapy, customer support, and human-computer interaction applications. In the therapy area, for example, it will give therapists quantifiable information concerning what emotional states the patients are in, complementing with clinical observations to offer deeper insight into patient development.

Chapter 6

Conclusion and Future Scope

6.1 Conclusion

The work focused on emotion recognition from speech using four machine learning models: CNN, RCNN, LSTM and Bi-LSTM. We trained and tested these models on two datasets, CREMA-D and TESS, that brought forward pretty different challenges. CREMA-D offered such more complex, real-world emotion data that led to lower accuracy around 40% because of its diverse and varied emotional expressions. On the other hand, TESS produced much higher accuracy (98-99%), but the models showed signs of overfitting due to the scripted nature of the recordings. We merged the two datasets, creating a more balanced training set to further limit overfitting, while still adequately exploring the emotional richness of CREMA-D. In general, this resulted in the improvement of accuracy as well as generalization performance across the models.

Among the models tested, LSTM proved to be the most effective in emotion recognition, implying the highest accuracy. The ability of LSTM to capture long-term dependencies in sequential data such as tone, pitch, and speech patterns makes it particularly suitable for emotion recognition from speech.

Its final application is a therapist's developed Web page for uploading the audio recording of sessions to this LSTM model so that it can process recordings and show progressions of emotions, enabling therapists to move in objective monitoring with data support, identifying patient emotional states and then tailoring therapy.

In conclusion, the combining of CREMA-D and TESS datasets using the LSTM model led to successful emotion recognition with real-world applications in therapy.

Its potential in the mental health domain and more can be further enhanced with future improvements in model accuracy and real-time emotion tracking.

6.2 Future Scope

Although promising in its results, there are a number of areas for improvement and further exploration:

(a) Improvement in Model Accuracy:

The generalization may be further improved by training this model with more diverse speech datasets comprising a variety of different emotions and demographics of speakers. More Advanced Architectures: Coupling with more advanced models like Transformer-based architectures or attention may also support more accuracy, particularly with the detection of subtle emotional cues.

(b) Multimodal Emotion Recognition:

Other Forms of Modalities: The speech features can be integrated with the other forms of modalities such as facial expressions or body language to give more representative and accurate understanding of the emotional state. Real-time Applications:

(c) Real-time Applications:

Emotion tracking in real time This system can be extended for real-time application in the sense that therapists could receive immediate emotion feedback during live sessions. Voice Assistants and Automated Customer Service: The emotion detection model can be used for voice assistants and automated customer service systems to better understand the user emotions and respond empathetically.

(d) Ethical Issues

Data Privacy and Security: As the system aims to process sensitive audio, securing strict privacy measures and acquisition of user consent is very vital. Generally, data anonymization techniques and secure protocols for data storage will be important to secure a users information. Granularity of Emotions

(e) Granularity of Emotions

Fine-Tune the Categories of Emotions: Existing categories of emotion might be extended or even refined to include more concrete situations, such as mixed emotions and transitions of complex phases of emotion, which may provide further insight for applications like therapy.

Chapter 7

Appendix

7.1 Appendix A: Acronyms and Abbreviations

7.2 Appendix B: Code Snippets

7.2.1 Data Preprocessing

```
# Map all possible labels to common emotions
emotion_mapping = {
    'ang': 'anger', 'dis': 'disgust', 'fea': 'fear',
    'hap': 'happiness', 'neu': 'neutral', 'sad': 'sadness', 'sur': 'surprise',
    'anger': 'anger', 'disgust': 'disgust', 'fear': 'fear',
    'happiness': 'happiness', 'neutral': 'neutral', 'sadness': 'sadness', 'surprise': 'surprise'
}
df['label'] = df['label'].map(emotion_mapping)
```

FIGURE 7.1: LSTM Data Pre-Proce ssing

7.2.2 Feature Extraction

```
# Function to extract MFCC features
def extract_mfcc(filename):
    y, sr = librosa.load(filename, duration=3, offset=0.5)
    mfcc = np.mean(librosa.feature.mfcc(y=y, sr=sr, n_mfcc=40).T, axis=0)
    return mfcc

X_mfcc = df['speech'].apply(lambda x: extract_mfcc(x))
X = np.array([x for x in X_mfcc])
X = np.expand_dims(X, -1)  # Add an extra dimension for input to LSTM
```

FIGURE 7.2: LSTM Feature Extraction

7.2.3 Model Training (LSTM)

```python
# Build the RNN model
model = Sequential()

# Add LSTM layers
model.add(LSTM(256, input_shape=(X_train.shape[1], X_train.shape[2]), return_sequences=True))
model.add(Dropout(0.3))
model.add(BatchNormalization())
model.add(LSTM(256, return_sequences=True))
model.add(Dropout(0.3))
model.add(BatchNormalization())
model.add(LSTM(128, return_sequences=False))
model.add(Dropout(0.3))

# Output layer (softmax for multi-class classification)
model.add(Dense(y_train.shape[1], activation='softmax'))

# Compile the model
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

# Train the model
history = model.fit(X_train, y_train, epochs=100, batch_size=64, validation_data=(X_test, y_test))

# Evaluate the model
y_pred = model.predict(X_test)
y_pred_classes = np.argmax(y_pred, axis=1)
y_test_classes = np.argmax(y_test, axis=1)
```

FIGURE 7.3: LSTM Model Training

7.3 Appendix C: Evaluation Metrics

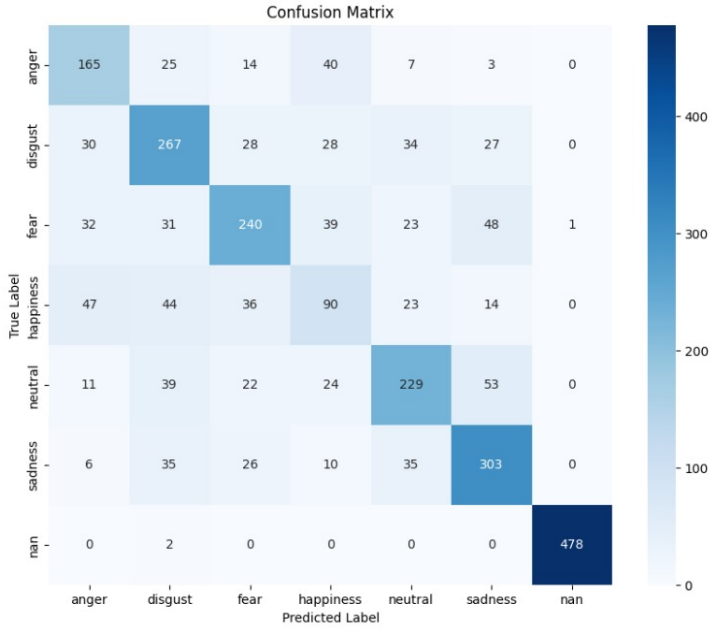

FIGURE 7.4: LSTM Evaluation

7.4 Appendix D: Web Application User Guide

- Accessing the Application:
- Open the Streamlit application:
- Uploading Audio Files:
- Emotion Tracking:

7.5 Appendix E: Limitations of the Model

- Limited Generalization: Performance may vary with speakers whose vocal characteristics differ significantly from the training data.
- Ambiguity in Emotional States: Difficulty in distinguishing emotions with subtle differences, like neutral vs. sadness.
- Dataset Bias: CREMA-D and TESS datasets may not fully represent real-world therapy scenarios.

Bibliography

[1] M.R. Amer, B. Siddiquie, C. Richey, and A. Divakaran, 'Emotion Detection in Speech Using Deep Networks.' ,SRI International, 201 Washington Rd. Princeton, NJ08540.

[2] C. Praksah , Prof. V. B. Gaikwad, 'Analysis of Emotion Recognition System through Speech Signal Using KNN GMM Classifier' ,International Journal Of Engineering And Computer Science ISSN:2319-7242 Volume 4 Issue 6 June 2015, Page No. 12523-12528

[3] D. Mamiev A.B. Abdusalomov, A. Kutlimuratov, B. Muminov, T. Keun Whangbo, 'Multimodal Emotion Detection via Attention-Based Fusion of Extracted Facial and Speech Features' , Sensors 2023, 23(12), 5475.

[4] F. Yu, E. Chang, YQ. Xu, HY. Shum, 'Emotion Detection from Speech to Enrich Multimedia Content.' Pacific-Rim Conference on Multimedia, 2001.

[5] K. L. Ong, C. P. Lee, H. S. Lim, K. M. Lim, T. Mukaida, 'SCQT-MaxViT: Speech Emotion Recognition With Constant-Q Transform and Multi-Axis Vision Transformer' , in IEEE Access, vol. 11, pp. 63081-63091, 2023. IJERT, Volume 8, Issue 08, NCICCT – 2020. DOI: 10.17577/IJERTCONV8IS08017.

[6] Gupta S., Kumar P., Tekchandani R.K., 'Facial Emotion Recognition Based Real-Time Learner Engagement Detection System in Online Learning Context Using Deep Learning Models.' IEEE Access, Volume 10, 2022.

[7] Monisha G.S., Yogashree G.S., Baghyalaksmi R., Haritha P. 'Enhanced Automatic Recognition of Human Emotions Using Machine Learning Techniques.' , IEEE Transactions on Affective Computing, 2023.

[8] Vijayanand G., Karthick S., Hari B., Jaikrishnan V. 'Emotion Detection using Machine Learning.' DOI: 10.1109/ACCESS.2022.3201234.

Biodata

Name: Aakash R. P.
Mobile No.:
E-mail: theertha.krishna2021@gmail.com
Permanent Address:

Name: Madhav Jay
Mobile No.: 6235392810
E-mail: madhavjay40@gmail.com
Permanent Address: Madhavam(H), Ettumanoor PO, Kottayam, 686631

Name: Theertha Krishna

Mobile No.: 9995857314

E-mail: theertha.krishna2021@gmail.com

Permanent Address: 1-C, B'Canti Celestial, Jawahar Nagar, Trivandrum, Kerala