RED-LSTM: real time emotion detection using LSTM

by

Fatin Ishrak

Abstract

The development of the Internet of Things and voice-based multimedia apps has allowed for the association and capture of several aspects of human behavior through the use of big data, which consists of trends and patterns. In the emotion of human speech, there is a latent representation of numerous aspects that are expressed. By mining audio-based data, it has been prioritized to extract sentiment from human speech. This capacity to recognize and categorize human emotion will be crucial for developing the next generation of AI. The machine will then begin to connect with human desires as a result. The audio-based data, such as voice emotion recognition, has not been able to produce results as accurate as those of text-based emotion recognition in terms of performance. For acoustic modal data, this study presents a combined strategy of feature extraction and data encoding with one hot vector embedding. When real-time data is available, LSTM has even employed an RNN-based model to forecast the emotion that captures the human voice's tone and signifies it. When predicting categorical emotion, the model has been assessed and shown to perform better than the other models by about 10%. The model has been tested against two benchmark datasets, RAVDESS and TESS, which contain voice actors' renditions of eight different emotions. This model beat other cutting-edge models, achieving approximately 80% accuracy for weighted data and approximately 85% accuracy for unweighted data.

Keywords: Machine Learning; Speech Emotion Recognition; Prediction; RNN; LSTM; Real-time Prediction

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

For real-time multimedia applications like voice control on wearables and online gaming, as well as many other services like real-time content delivery, video conferencing, and more, internet traffic for multimedia has increased tremendously [8]. Rich sources, including integrated audio, video streams, and texts, make up multimedia systems. Numerous human conversations containing a vast amount of information are being signaled as a result of the internet's explosive expansion, and this signaling is especially having an impact on how human behavior and embedded emotions are expressed, with a focus on voice-related characteristics. Real-world thought and behavior are influenced by emotions in such a way that they are dependent on actual life circumstances. Humans connect in a special way and occasionally display a variety of emotions mixed together [5]. Not just this, but these emotions which may be simple or complex have an impact on how we move, think, act, and behave, as well as other factors [3]. Therefore, the amount of study being done in this area is growing daily due to the ability to identify, capture, and use human emotion through digital sources of multimedia.

Understanding emotion is useful in many ways. When we recognize an emotion, we come to understand that emotion plays a role in how we interact with others and how we make decisions [3]. Although scientists and psychologists are becoming more interested in this topic, a precise prediction of how emotions influence human cognition has not yet been produced. They concentrate on the production of emotions from various angles, including cognitive, neurologic, and social science perspectives [3]. However, it has been discovered that psychological state and perspective on a circumstance work together to generate emotions. For a neurologist, the distinction between positive and negative emotion is made based on the change in neuronal simulation density across time. The effectiveness of this division, however, is debatable because the valence tag, or the positive or negative of an emotion, depends on the circumstance and, in most cases, calls for a more in-depth and nuanced interpretation.

Recent research has shown that low-level auditory features such as spectrograms, MFCC, and fundamental frequency (F0), in addition to other deep-learned features, generate high-level statistical methods for emotion recognition [11] [17]. However, deep neural networks and sequential models like LSTM have been created primarily for audio emotion recognition. The capacity of RNN models to model sequential data while preserving complicated information has led to their widespread application. Additional dense layers for the attention mechanism in the LSTM layers

allow it to precisely identify the emotion that is concealed in human speech. The Bag-of-Audio Words (BoAW) mechanism for embedding structure is well known for its success in feature extraction for dimensional emotion detection. However, the model's robustness in terms of finer-grained categorical emotion, such as happiness, sadness, or disgust, has not been tested to the point where it is used by all researchers [1]. Recent studies have implemented numerous model architectures that can comprehend conversational context, greatly improving their ability to predict emotional states.

1.1 Research Problem

As the emotion identification is to be done from the audio dialogue, there are a few concerns that need to be addressed. The difficulty of extracting emotion from audio data is stated below.

- **Several different forms of emotions:** Mathematical prediction of different types of emotions is not possible with current technology. Additionally, no mechanism has been developed to forecast mixed emotions and assess the prediction [5].
- **performance issues:** Compared to textual feature extraction, audio feature extraction for emotion detection has some performance restrictions. Additionally, there are complementing indications for identifying emotions in the text-based and audio-based feature modalities [17]. It has not yet been determined why audio-based sources of emotion are so inaccurately detected.
- **Speaker journaling:** Depending on who was speaking, the section of the voices required to be separated from the discussion and given into the mood prediction model.

1.2 Research Objective

In order to address the issue, we divided our contributions to extract emotion from audio-based data into four main categories.

- **First,** To include superior audio qualities that distinguish eight major emotions, such as happy, sad, neutral, and quiet, we developed a Natural Language Processing-based encoding technique called the one-hot encoding approach. They are thoroughly covered in the chapter on datasets.
- **Second,** We employed a suitable attention method that provided the best padding with the feature representation inputs for the emotion detection model such as LSTM together with one-hot encoding.
- **Third,** In order to achieve a satisfactory real-time speech emotion recognition accuracy, we improved the performance of the model of each component using our approach, modifying various parameters for emotion prediction. This prompted us to assist in patterning the conversation's context and tone in order to accurately forecast the emotion class.

- **Finally,** Using the system microphone, we developed a real-time emotion prediction system and validated the methodology for usage in practical applications. This will aid in capturing the range of emotions used by conversation starters such as call centers run by humans or machines and the healthcare system that are automated. We added a summary log so that it can detect the final emotion based on emotion chunks in an entire session so that the theme can be understood.

The remainder of the essay is structured as follows. The basis of the theories we employed and relevant research was covered in the part that followed. We provided a thorough description of the LSTM model's architecture as well as the model we employed for data processing in chapter 3. In Chapter 4, the methods for processing the data technically are presented and the data is designed to be understood through visualization. The use of the LSTM model and real-time emotion detection is then demonstrated in Chapter 5. In chapter 6, we examined the output of the model's real-time speech emotion recognition and gave the results. The concluding section of chapter 7 is where future improvements and a summary of the entire effort are presented.

Chapter 2

Literature Review

A variety of scientific disciplines, including psychology, natural language processing, cognitive science, and machine learning, have contributed to the research on emotion detection [2]. When it comes to working with machines to sense human emotion, artificial analogs like emotional intelligence are crucial. The normalization of social connections with productivity is crucial [6]. The amount of research done on emotion detection to date is enormous and is based on pertinent data from numerous datasets that provided strong and convincing evidence of its impact on various industrial fields, including healthcare, human resource management, and artificial intelligence [13] [23] [24] [26] [20]. The low-level descriptors (LLD) is a time-tested method of predicting emotion that functions as frame-based feature extraction and now incorporates the collection of utterance-level data [12]. It is used as an input to a classification model with a regression technique. These studies emphasize the use of robust models and feature engineering in many types of emotion detection. In the current research, deep learning of the features and the extraction of audio features are both done using handcrafted methods. The major goal of the emotion detection model was to test out multimodality, including text, audio, and visual components, as well as contextually based emotion prediction using the classifier known as Support Vector Machines (SVM) and other deep neural networks. However, given the poor ability to forecast higher variation than only fundamental emotions [19] [17] and the lack of conversational context use [19], there is a great deal of room for future invention and improvisation over the proposed research in terms of accuracy and interpretability. The performance of the audio-based technique is subpar in comparison to text-based emotion detection [16].

2.1 Background

In the beginning, shallow hand-crafted techniques are used to extract features. Later Algorithms use a variety of statistical functions, including mean, range, variance, and coefficients of linear regression. These temporal characteristics were established, allowing the deep learning method to reveal the feature representations. Such a deep-learned characteristic was used by Adikari et al. [25] to recognize emotions. In a similar work, researchers look for the ideal pitch data to identify emotions in audio data. According to Mel-scale spectrograms, the pitch information is damaged. In order to perform better, the research [22] [12] makes use of linearly spaced spectrogram characteristics. It has been demonstrated that using statistical methods in

deep learning models results in greater performance than using hand-crafted features [12]. Motivated by a method of Natural Language Processing The classification of audio events and other activities has been accomplished successfully using One-hot Encoding feature extraction [10]. Mel-Frequency Cepstral Coefficients (MFCC) are used in this study as low-level features with a random sample of audio words. The parameters of valence and arousal, which represented positive and negative emotions respectively, were integrated with those characteristics and used as input in a Support Vector Regression model that predicted the emotion. They mentioned the intensity of the emotion as well. Despite this, the evaluation of the feature representation has been limited to just the arousal and valence parameters for recognizing emotions following this strong interpretability.
The majority of the research has been devoted to developing models for emotion detection. The researchers studied the Interactive Emotional Dyadic Motion Capture (IEDMC) dataset because there was limited publicly available data. Sequence modeling techniques have made advantage of the audio and text analyses. The evaluation of the dyadic discussion in the IEMOCAP dataset, which comprises of Multimodal Emotion Lines Dataset (MELD), which covers an average of 5 participants in the conversation, was the key problem. The models are divided into classifier-based deep learning models, context-dependent or independent predictors, and multimodal emotion detection models. The majority of efforts have been performed with multimodal information in order to produce a viable solution. In literature, the idea of using multimodal features like textual, visual, and auditory modalities for their complementing information is created into a rich feature representation. A CNN-based architecture in the study [19] illustrates the influence of voice and text transcription in a speech to identify the emotion. This study compares the effectiveness of spectrogram features based on audio modality with Mel-Frequency Cepstral Coefficients (MFCC). The speech-to-text translation process may lose information due to the limited feature representation capability of text embedding. Furthermore, this research's accuracy has significantly improved by integrating the text and audio modalities. Modern multimodal emotion recognition technology has been demonstrated in a publication [14]. They employed CNN single layer with textual characteristics and pre-trained word embeddings. OpenSMILE [4] toolkit's audio features have been extracted, and 3D CNN has been employed for the visual modality. Although there is still a sizable difference between the three modalities, multimodal techniques for emotion recognition are more accurate than unimodal ones [16].

2.2 Related Works

The main model for emotion recognition today uses deep learning techniques and tree- or distance-based machine learning algorithms. For instance, in speech-based emotion detection, the audio feature differentiates between high and low to activate emotion for the initial top-level split. The research [3] employed a decision tree classifier that was a hierarchical one that could leverage prior knowledge. Next, run a series of classifiers, including Gaussian Mixture Models, Linear Discriminant Analysis, and Support Vector Machine to cascade and split Support Vector Machine for the best performance in the baseline of accuracy. Convolutional and recurrent neural networks, among other deep learning techniques, have been applied [12].

The study in [7] built a deep learning (DL) based prediction system that forecasts segment level on emotion probability distribution and input into a single hidden layer network established for utterance level emotion detection in an Extreme Learning Machine (ELM). Based on emotion recognition for a small size split of the training set, the Support Vector Machine, the ELM, and DL as a stack are effective and outperform. The convolution neural network has been proposed along with the sequence modeling strategy, using combined feature flow from the method Mel-Frequency Cepstral Coefficients (MFCC) and spectrograms [19]. According to the study [9], a novel strategy merged CNN-based features from sequential data and fed them as input into an RNN.

The model based on RNN used a memory network combined with an attention mechanism and performed satisfactorily to extract historical features of the dialogue for identifying the emotion, whereas the majority of studies were based on gathering emotional information from individual utterances [21] [22]. Researchers currently choose RNN models over traditional Hidden Markov Models for sequential data modeling tasks such as speech recognition and emotion detection because they handle incoming data through the recurrent structure (HMM). Additionally, the LSTM version of the most recent RNN performs satisfactorily in terms of identifying long- and short-term conversational notions [27]. Due to their capacity to add memory cells that track the memory state in sequential data, RNN models in emotion detection of multi-party communication's context have therefore acquired significance in order to better utilize contextual knowledge of the conversation [28]. Human nature is typically influenced by emotion that arises from conversational context; therefore, a model must have the ability to recognize complicated emotions while preserving accuracy and ensuring improved conversational context use. In order to anticipate emotions, Poria et al. [15] used contextual data from the same speaker's subsequent utterances. Recently, by removing emotionally salient information from utterances and adding conversational context to the layer, a considerable accuracy boost was realized.

An attention-based weighted pooling method combined with a bidirectional LSTM was utilized in the study [11] to help the network focus on and capture emotionally significant passages of speech. Using an RNN memory network with a multi-hop attention mechanism that incorporates interpersonal and self-influence into the global memory of the dialogue, this method in [14] learned an emotive summary of the situation. The DialogueRNN and its variations (BiDialogRNN, BiDialogRNN with attention) are state-of-the-art models that have independent Gated Recurrent Units and are used to collect context information while taking into account the global context of the discussion, emotion state, and speaker status [18]. Performance of this model has been assessed for a trimodal scenario and textual modality, however individual audio modality has not yet been established.

In comparison to text-based modality, one of the main constraints on feature representation in the existing work is the low precision of the audio modality. Additionally, there is a clear lack of understanding of how audio embedding techniques, which are used to classify emotion, have an impact on NLP capabilities. Superior performance for emotion detection can be obtained by using RNN models and the appropriate attention mechanism in conjunction with the contextual information flow in the discussion, which is the realization of this research study. In order to overcome the aforementioned constraint, we tested a new audio feature extrac-

tion technique employing an RNN-based LSTM and three energy-based approaches (Root Mean Square (RMS), Zero Crossed Rate (ZCR), and Mel-Frequency Cepstral Coefficients (MFCCs)) to enhance emotion performance from human speech for real-time.

Chapter 3

Model Architecture

The goal of this research is to use artificial neural networks to create a real-time emotion recognizer of human speech. Because the gathered data comprises a range of values on various scales, the time series must first be fitted and normalized in order to enhance network training. There must first be a stage of preparation for the data. After that, a data sample is trained using a serial-parallel architecture. After the training operations, the serial-parallel architecture is transformed into a parallelized network to carry out prediction tasks.

3.1 Long Short Term Memory

Implementing the PCA parameters manually is very difficult and almost impossible to optimize. This research mitigates the problem using a more complex model so that the model can compute each past data point's significance and provide optimized predictions. Thus, the implementation of the Long Short Term Memory (LSTM) model provides the weight updation while training the ML model.
LSTM as an RNN provides the scope to work on data sequences and ease learning by retaining only the relevant information from the time scope. The extracted information from the network that is learned by the model is added to a memory that gets updated after each timestamp based on the significance of the new information to the sample.
The model implements the LSTM cell each with three gates illustrates as the input gate, the output gate, and the forget gate. The three gates combinedly perform to learn the weights and determine the ratio of a current data sample to be remembered and past learned context should be forgotten. The cell state C_t represents the short-term and the long-term internal memory of a cell.

3.1.1 Input Gate

The input gates have been implemented to select the new information to be added and stored in the current C_t. A *sigmoid* function is implemented to reduce the input vector (i_t) values.

$$i_t = \sigma(W_i.[h_t - 1, x_t] + b_i) \tag{3.1}$$

After that, a *tanh* function modifies each value between $[-1, 1]$ $(C-t)$. An element-by-element matrix has been multiplied by i_t and C_t which represents the information that requires to be added to the current cell.

$$\simeq C_t = \ tanh(W_C.[h_t - 1, x_t] + b_C) \tag{3.2}$$

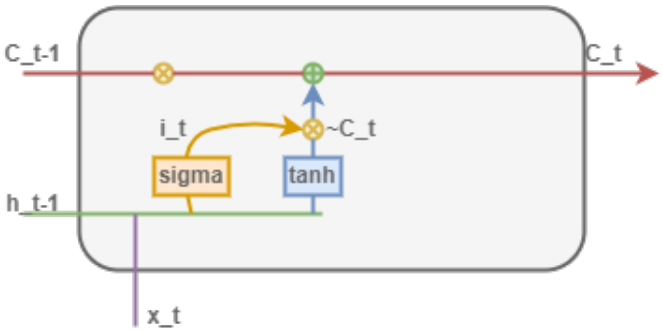

Figure 3.1: Input Gate

3.1.2 Output Gate

To control the output flowing to the next cell the output gate has been implemented. The output gate consists of a *sigmoid* function and then to filter the less important information a *tanh* function has been implemented. By this technique, the information that needs to pass through is kept. The output (o_t) is calculated by the following equation.

$$o_t = \sigma(W_o.[h_t - 1, x_t] + b_o) \tag{3.3}$$

And the required (h_t) is as follows.

$$h_t = \ o_t * tanh(C_t) \tag{3.4}$$

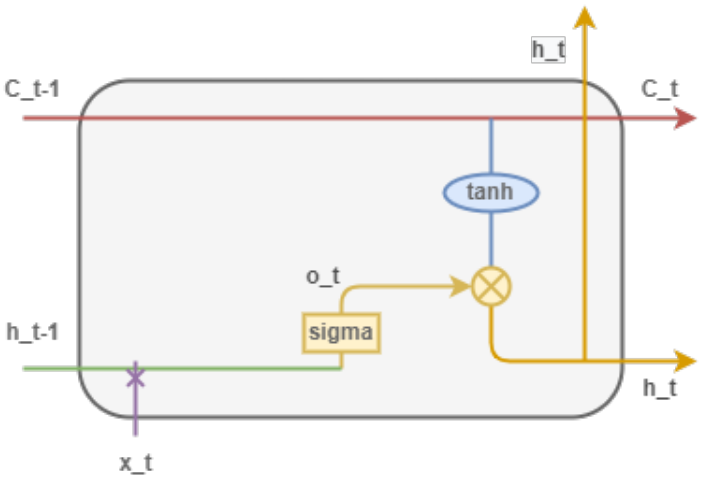

Figure 3.2: Output Gate

3.1.3 Forget Gate

The implementation of the forget gate confirms which information is to be forgotten by the model. The filtered information that the model can recognize as less important has been thrown away with this implementation of the forget gate. Mathematically, the forget gate has been implemented with the *sigmoid* function such as the output value range between $[0, 1]$ from the $C_t - 1$ state. 1 indicates the complete passing value and 0 defines the completely filtered-out values. The following equation has been implemented for the forget gate.

$$f_t = \sigma(W_f.[h_t - 1, x_t] + b_f) \tag{3.5}$$

The figure illustrates the functionality of the forgetting gate in an LSTM cell.

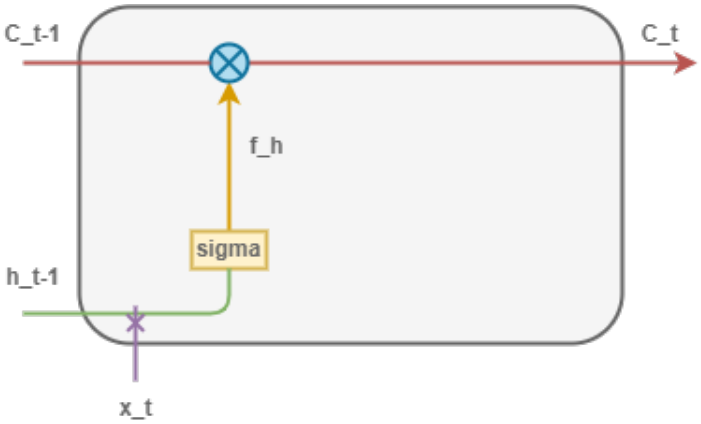

Figure 3.3: Forgate Gate

3.2 Data Modelling

An overview of the dataset and the design of the data processing for our model are given in this section. The data processing architecture employed in the LSTM

model is depicted in the following picture.

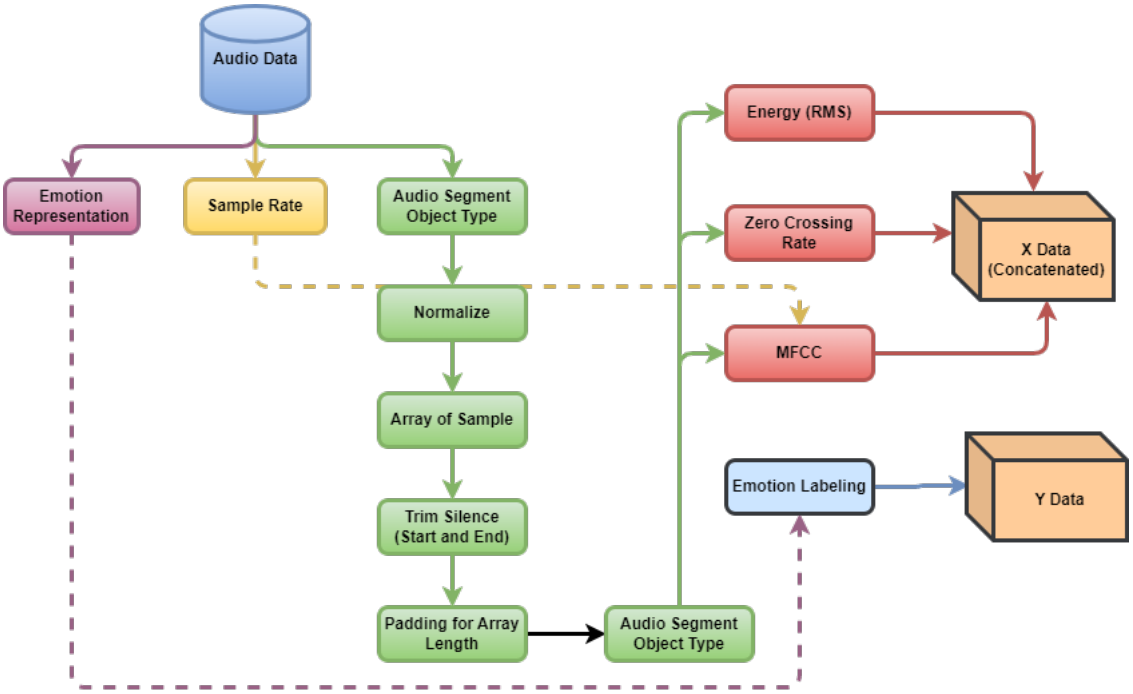

Figure 3.4: Architecture of Data Processing

Each audio track has been taken from the dataset. In RAVDESS, the emotion representation of the dataset is denoted by a number, such as 03 for happy, while for TESS, the file name serves as a clear indication of the emotion. The second subprocess is the rating of samples. The parameter of the number of audio samples per second is included. TESS uses 22.5 kHz, whereas RAVDESS uses 48 kHz. The audio is then divided into segments as an object type of data, with an instance of Audio Segment that is imported into an object by the library called "AudioSegment." The process then returns to normal. The item was normalized to +5.0 dBFS. The item was then converted into a sample array, which is one of the crucial steps in the preprocessing. Next, the data is cleaned up to remove any unnecessary information and prevent beginning and ending silence from biasing our model. After that, padding was added to every audio file such that it contained equalization that was the same length across the board. The process of noise reduction is then completed. The features were then taken from the data to begin the machine-learning process. We extracted three characteristics from our dataset:

- Energy - Root Mean Square (RMS)
- Zero Crossed Rate (ZCR)
- Mel-Frequency Cepstral Coefficients (MFCCs)

After combining all the features and emotions and saving them as supervised learning labels in the test data, the data has been stored as a train set.

Chapter 4

Dataset

RAVDESS and TESS are two of the datasets we chose for this paper. 24 professional actors, including 12 men and 12 women, are included in the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) data. The performers delivered the lines in a neutral North American accent while using two lexically similar statements. Two levels of expression intensity are produced: one is normal and the other is strong with an additional neutral expression. The Northwestern University Auditory Test No. 6 served as the inspiration for the Toronto Emotional Speech Set (TESS) (NU-6; Tillman and Carhart, 1966). 200 target words make up the dataset. The recording uses a spoken carrier phrase, and when it ends, two women play seven different emotions: neutral, happy, sad, angry, fearful, disgusted, and pleasant. There are eight emotions in the RAVDESS dataset, neutral, calm, happy, sad, angry, afraid, disgusted, and astonished. The TESS dataset consists of 2800 files with 2 actors multiplied by 200 phrases multiplied by 7 emotions, whereas the RAVDESS datasets have 1440 files with 24 actors multiplied by 60 trials per actor.

4.1 Data Preprocessing

We extracted the sample using the *AudioSegment* module of the *pudub* library in order to observe the data. The audio data was then visualized using the *librosa* library and an array is added as the sample. The time scale of the actual audio is on the $x - axis$ of the data visualization, while the loudness is on the $y - axis$. The actual audio is very faint, as indicated by the $y - axis$ range. This could conflict with the validity of the feature extraction with interference.

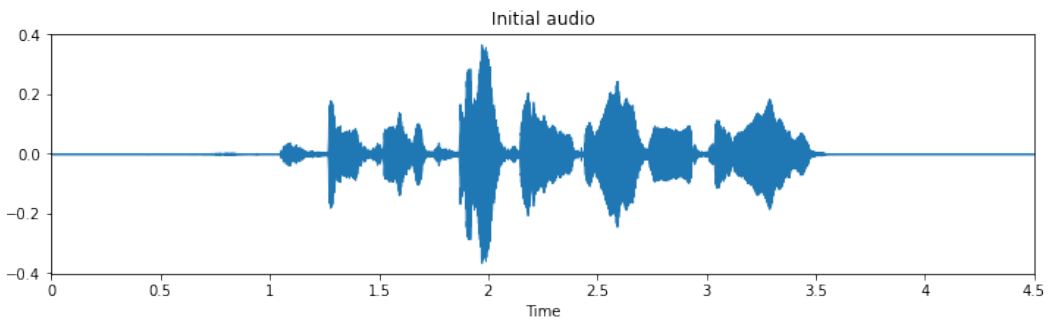

Figure 4.1: Actual Audio

Here, we normalized the sound to $+5.0dBFS$ using the $effect$ module of the *pydub* library. After that, the normalized track became a *numpy* array. The normalized audio is shown in Figure 4.2, and the significant volume increase is indicated on the $y - axis$.

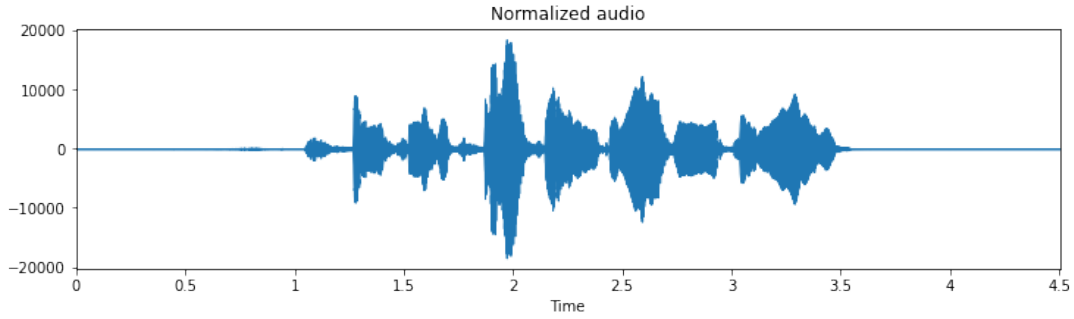

Figure 4.2: Normalized Audio

The trim, which was carried out with the $effec$ module, appears as follows and reduces the flat line's starting and finishing ends while removing extraneous data. The trimmed audio is displayed in the following figure.

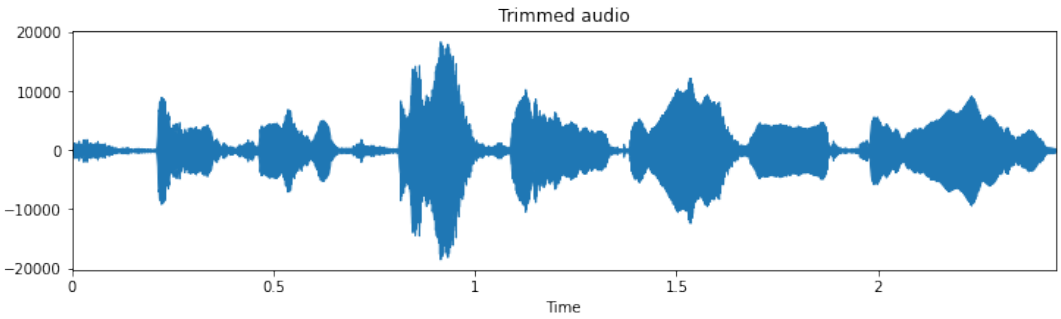

Figure 4.3: Trimmed Audio

For length equalization, padding has been provided to the right side. The computed maximum audio duration is 243200. The padded audio can be seen in the following graphic. The *numpy* has been used for padding.

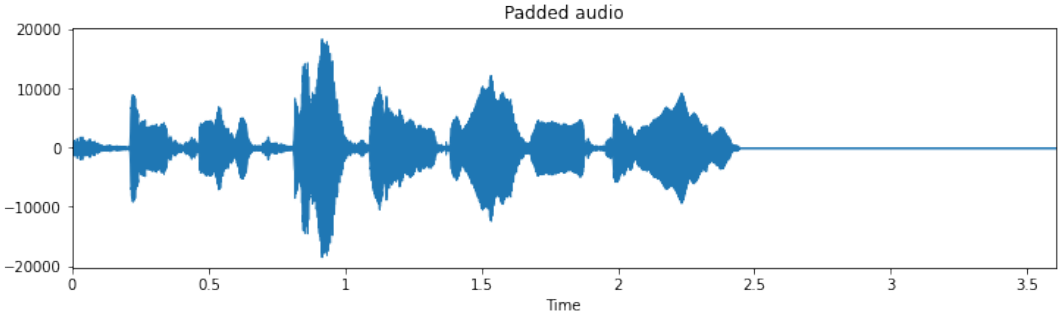

Figure 4.4: Padded Audio

Although none of the datasets include any noise, the *noisereduce* package offers a consistent stamp. The following graph demonstrates that the $y-axis$ has a constant value. This data is stored as the final data in an array.

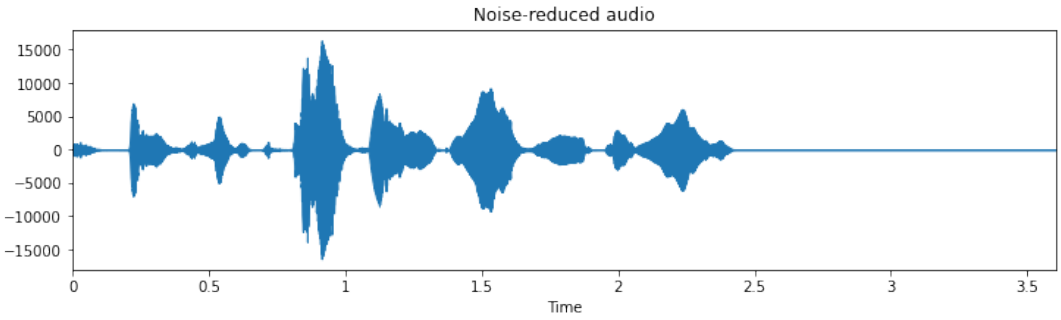

Figure 4.5: Noise-reduced Audio

4.2 Features Extraction

The *librosa* has been used to extract the features for speech emotion identification. The features collected in this case include $Root\_Mean\_Square(RMS)$, $Zero\_Crossed\_Rate(ZCR)$, and $Mel-Frequency\_Cepstral\_Coefficients(MFCCs)$. The hop_length is set to 512, and the $frame\_length$ is assumed to be 2048. Every 2048 sample was examined, and 4 sequential feature values

$$(2048 \ / \ 512 = 4)$$

were obtained. As a result, 475 sequential values, one for each feature, are returned for the 173056 entire lengths of the audio sequence plus one for the last sample

$$(173056 + 1/ \ 512 = 339)$$

. The values of the features are depicted in the following table. The Root Mean Square is used to calculate $Energy$.

Feature	Shape
Energy	(1, 339)
ZCR	(1, 339)
MFCCs	(13, 339)

Table 4.1: Feature Values

The observation of a sample of data described before marks the end of this section. Additionally, each feature has 339-time steps, proving that the data is uniform in duration and quality.

4.3 Combining Data

When it comes to how emotions are represented, the two datasets are different from one another. The audio file name in the RAVDESS database has a 7-part

numerical identification. On the other hand, the file name of the emotion name in the TESS database is a string. A function that can represent a value from a string that is comparable to another value in a database has been implemented to solve the problem. We created a function that can do the $n = n - 1$ procedure to express emotion because the classification will accept models with values starting at 0. The maximum sample length has then been determined for padding using *librosa*, and all of the values have been inserted into an array where the maximum sample count is found.
Each file is then normalized and turned into a *np.array* of samples after all the data has been analyzed. The trimming and cushioning were done after that. The noise reduction process is then complete. RMS, ZCR, and MFCC were then extracted after that. The combining function was then appended separately to each feature in the dataset after being applied to the entire dataset.

4.4 Data Preparation

The features' shapes have been modified to be symmetrical and arranged in a three-dimensional array in order to prepare the data for training in the LSTM model. The features have all been combined into one variable. Using the *Keras* library, the Y has been changed to a 2D shape. Following that, the data was divided into a train, test, and validation set. In order for *y_train* and *y_validation* to remain adjacent to the *test* set, one-hot vector encoding was used to convert them. The shape of each feature in the combined dataset is shown in the following table.

Feature	Shape
Energy	(3812, 951, 1)
ZCR	(3812, 951, 1)
MFCCs	(3812, 951, 13)

Table 4.2: Feature Shapes

The dataset was then divided into three sections: *x_train*, *x_val*, and *x_test*. Their shape is depicted in the accompanying illustration with *y_val* values.

Data	Shape
x_train	(3335, 951, 15)
x_val	(331, 951, 15)
x_test	(146, 951, 15)
y_val	(331, 1)

Table 4.3: Array Shape

We will train our model with LSTM layers using this dataset in the following chapter.

Chapter 5

Implementation

For data preprocessing, we used the scikit-learn module, Keras, and Tensorflow, with Keras acting as the front end of the machine learning. The data we used in this research includes a variety of parameters.

5.1 Model Justification

The model has been run using the Keras library. In this study, a dense layer with an output of 8 nodes implements two hidden LSTM layers, each with 64 nodes. One of the emotional representations is represented by each node. We utilized the "softmax" function for the activation. The "RMSProp" has used the optimizer with its default setting. We settled on 23 as the batch size because it is a factor of all the samples in the dataset and produces the best results. The system-generated graph of the nodes indicating the shape in each LSTM layer is depicted in the following figure. The Keras library was used to generate the output.

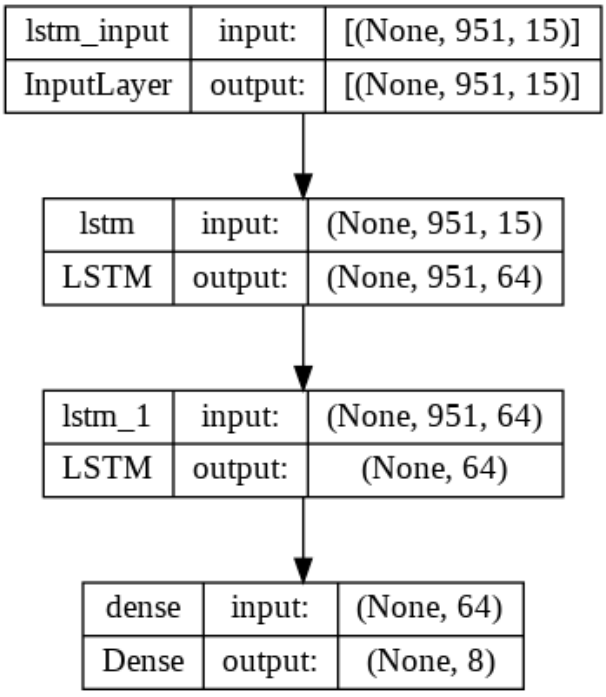

Figure 5.1: LSTM Layers

5.2 Model Parameters

The model is set as a *Sequental* with $softmax$ activation in the dense layer because the audio data is sequential and LSTM is used for sequential data learning. The batch size has been set up from scratch. The training data is evaluated as having a maximum level of valid categorical accuracy, and it is saved in $HD5$ format for real-time emotion recognition. After 100 iterations without improvement, the learning rate was decreased by a factor of 0.1. We used *categorical_crossentropy* and *categorical_accuracy* matrices to compute the loss function, utilizing $RMSProp$ as the optimizer. The weights have been saved after 340 iterations of the model. The model and its parameters are depicted in the following table.

Model: Sequential

Layer (type)	Output Shape	Param
LSTM	(None, 951, 64)	20480
LSTM_1	(None, 64)	33024
Dense	(None, 8)	520

Table 5.1: Training Parameters

5.3 Real Time Speech Emotion Recognition

Based on our trained model, we implemented real-time speech emotion identification in this study. The implementation will create a temporary *.wav* file and record audio input from the device's microphone. The recorded audio will be preprocessed and returned in the audio file along with a distribution of the speech's emotional content. We utilized the *pyaudio* package to record audio. First, $Keras$ and $TensorFlow$ are loaded and the pre-trained model is compiled. Following data processing, each array's features were retrieved from the data. The MFCC has been set at 13. The X has been transformed into a 3D array after concatenation. The emotion list is defined to allow for legible output. The "is silent" function has been used so that the audio recording would automatically pause and form an input after a period of silence. This LSTM model implementation offers real-time emotion prediction as speech emotion recognition out of the audio output. The device's sound card is used to record the audio. The session begins after *pyaudio* joins the input channel. The audio signal is then recorded using *pyaudio* and *wave* if there is no quiet. The recorder will stop after a set amount of time and begin processing, after which the subsequent recording will begin and be preprocessed.
An array of 8 emotion probabilities is returned when the *model.predict* is invoked. Predictions, for instance, could be written as [array([p_neutral, p_calm, p_happy, p_sad, p_angry, p_feaful, p_disgust, p_suprised], dtype=float32]]. Following that, the predictions are made into a brief visualization and saved in a list. The expected result has been displayed using the *matplotlib* package. If there is quiet for more than two seconds, the session will end automatically, and the summary will show the total time of the session while also terminating the input connection. The sampling rate, in this case, is 24414 while the chunk represents the hop length. The sample

size is the same as the model, and the 32-bit audio recording format only accepts input from a mono channel.

Chapter 6

Result

6.1 Model Evaluation

The display of the loss and categorical accuracy values that are produced throughout the training phase has been used to evaluate the model. A confusion matrix has also been created to show the percentage of correct predictions for each emotion in the validation and test sets. Finally, test and validation sets have shown the model's prediction accuracy rates for each emotion. The following figure illustrates the training and validation loss.

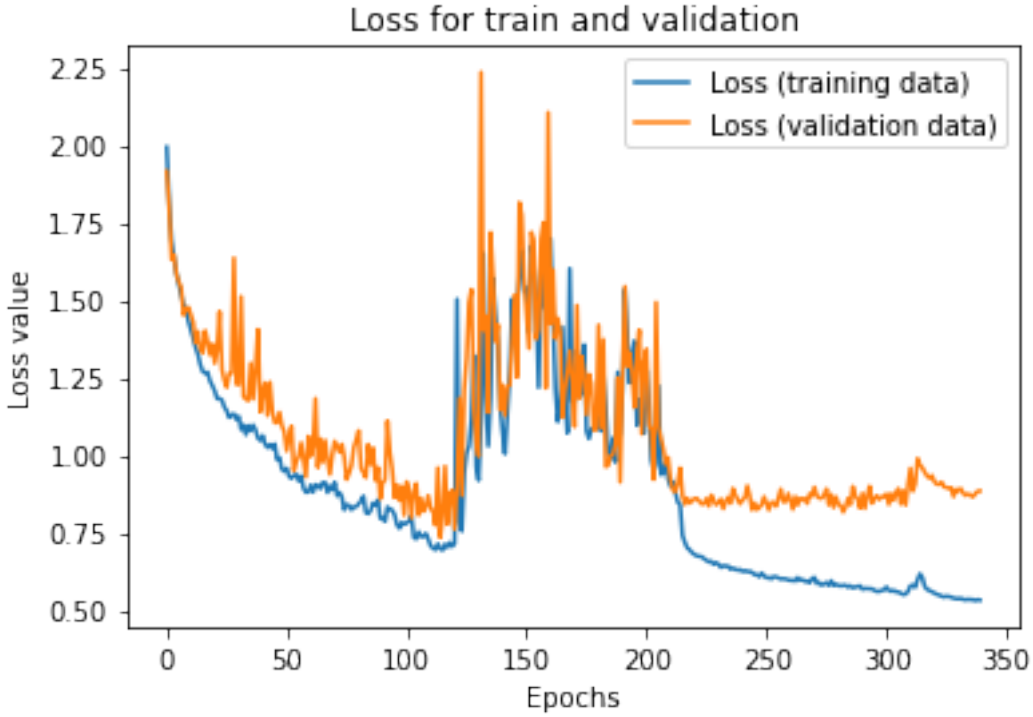

Figure 6.1: Training and Validation Loss

From the figure, we can see the training loss is less than the validation set after 200 epochs. In the next figure, the accuracy of the model is shown.
From the graph, it is visible that the model has achieved 81% accuracy while the training phase.

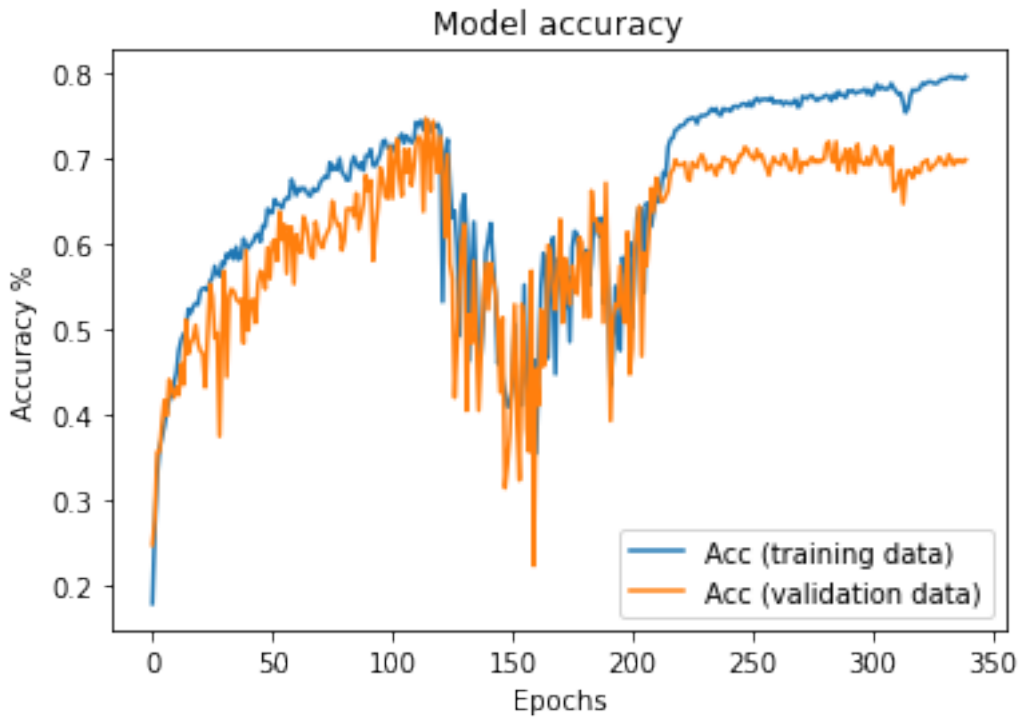

Figure 6.2: Accuracy

6.1.1 Validation Set Evaluation

The validation set is evaluated with the confusion matrix, where the emotions are predicated on the validation set. The confusion matrix is shown below. From the matrix, it is clearly visible that in most of the cases, our prediction is correct. When the emotion is happy our model performs quite well, then sad and angry were the most accurate.

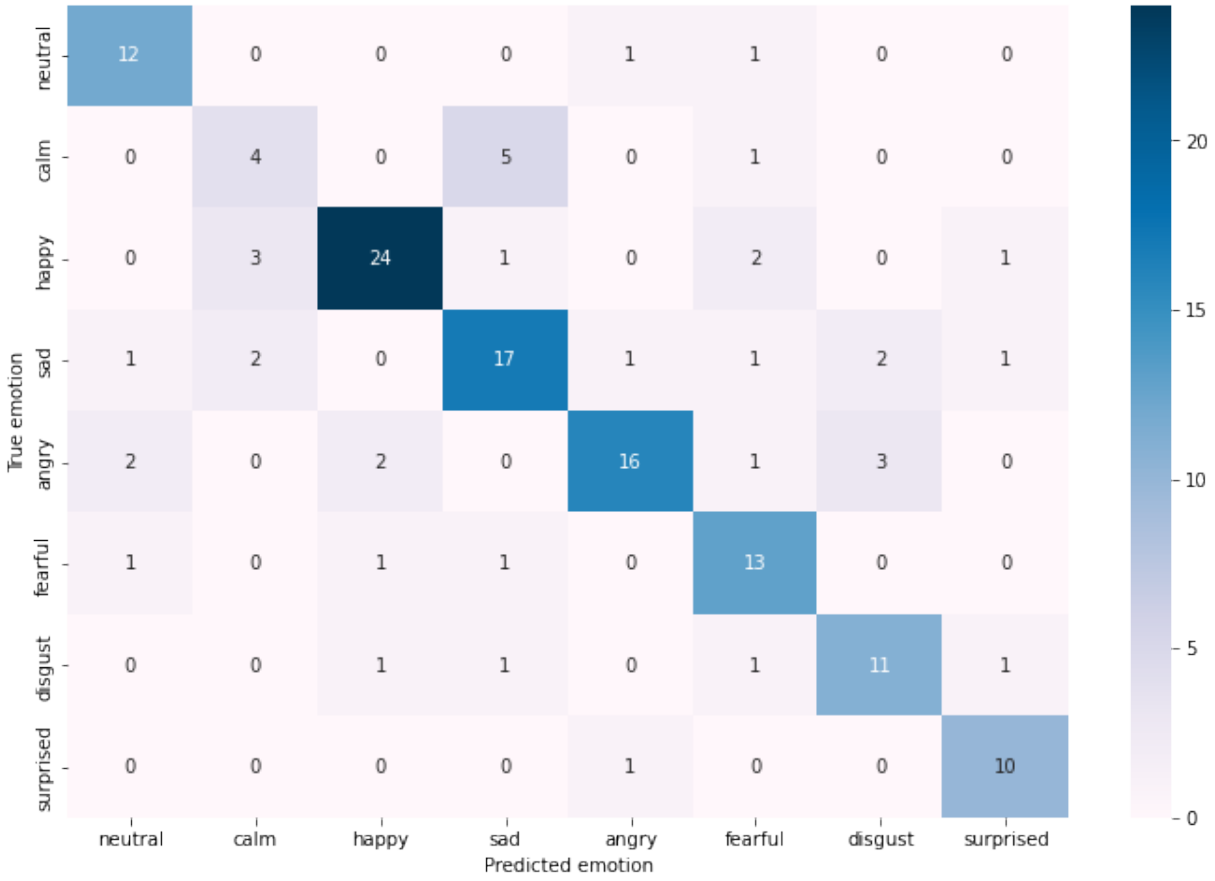

Figure 6.3: Confusion Matrix for Validation Set

The following table shows the accuracy of each emotion predicted in the validation set. Here, it is visible that our model achieved 91% accuracy on surprise then the neutral is the second best case. Our model did not perform well when the emotion is calm.

Emotion	Values
Neutral	0.8571
Calm	0.4000
Happy	0.7742
Sad	0.6800
Angry	0.6667
Fearful	0.8125
Disgust	0.7333
Surprised	0.9091

Table 6.1: Validation Set Accuracy for each Emotion Type

6.1.2 Test Set Evaluation

The confusion matrix is used to evaluate the test set's emotions and serve as the basis for those emotions. The following image represents the confusion matrix. The

matrix shows that our model accurately predicted the happy feeling, followed by the sad, angry, afraid, and neutral emotions, in that order.

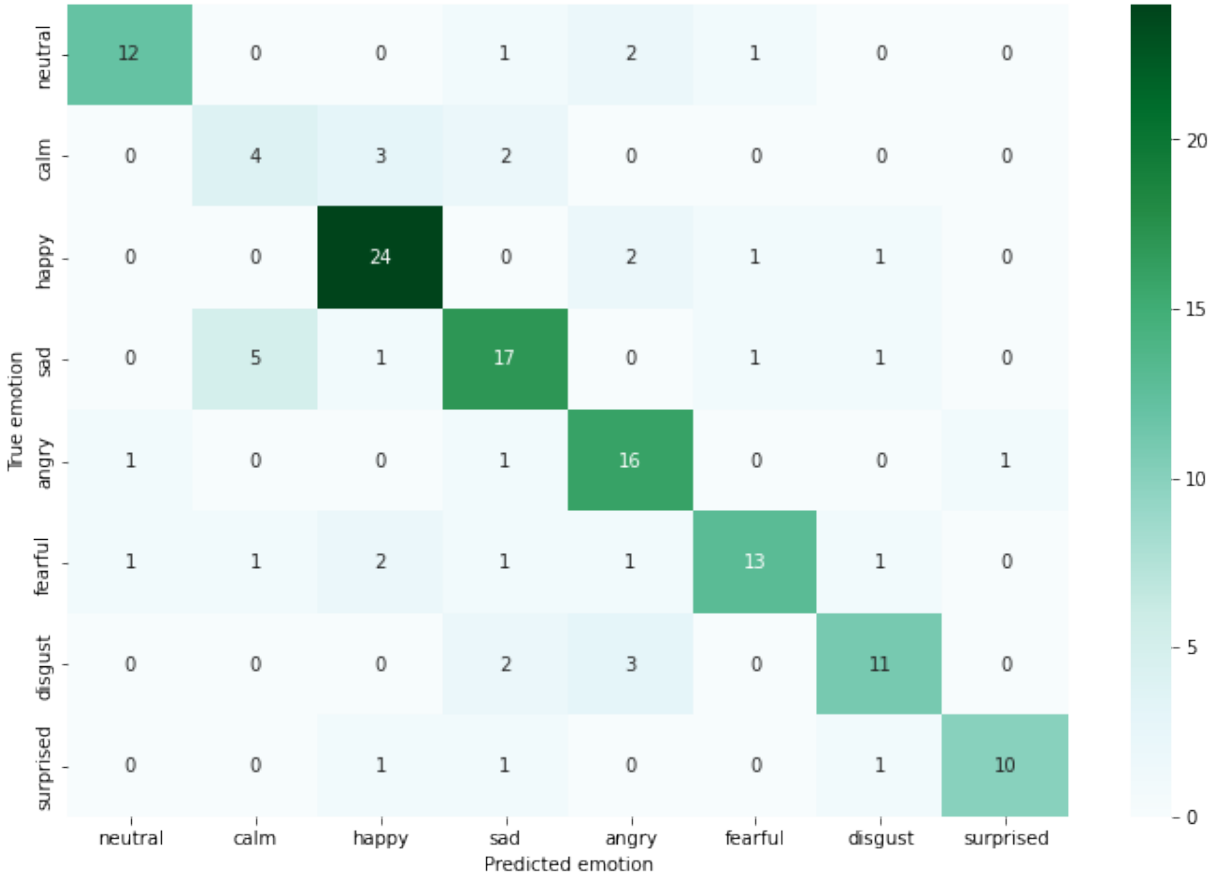

Figure 6.4: Confusion Matrix for Test Set

The accuracy for each emotion in the training set is shown in the following table. Here, the cheerful accuracy has a 95% success rate, and each emotion's learning rate is encouraging.

Emotion	Values
Neutral	0.8571
Calm	0.4000
Happy	0.7742
Sad	0.6800
Angry	0.6667
Fearful	0.8125
Disgust	0.7333
Surprised	0.9091

Table 6.2: Test Set Accuracy for each Emotion Type

6.2 Real Time Performance Evaluation

We collected the input for the real-time evaluation using the device's microphone. We tested the model using three different emotional tones and got the intended outcome. The graph in the next image depicts what happens when the input was purposefully set to a heavy voice with a gloomy mood and produces the desired outcome.

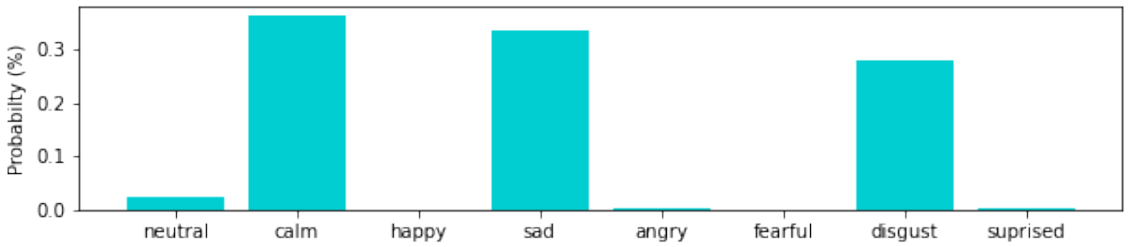

Figure 6.5: Calm Voice Input

Next, we purposefully maintained our composure and tried not to feel anything else. The forecast shown in the accompanying figure is accurate in this instance.

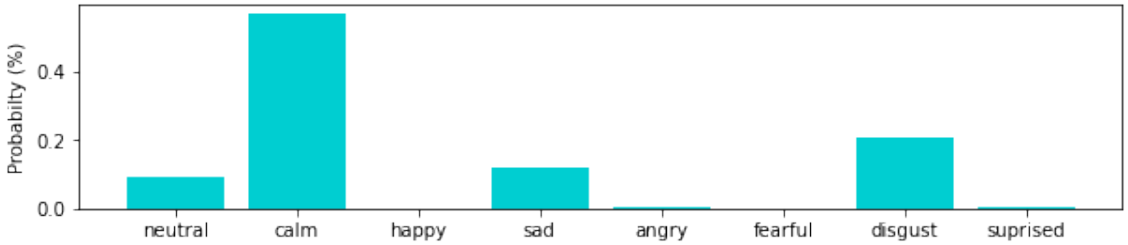

Figure 6.6: Correct Prediction

We inputted relatively typical speech for the final exam and let out a single shout to assess how well we did in a variety of scenarios. The output of the model along with the forecast is displayed in the ensuing graph.

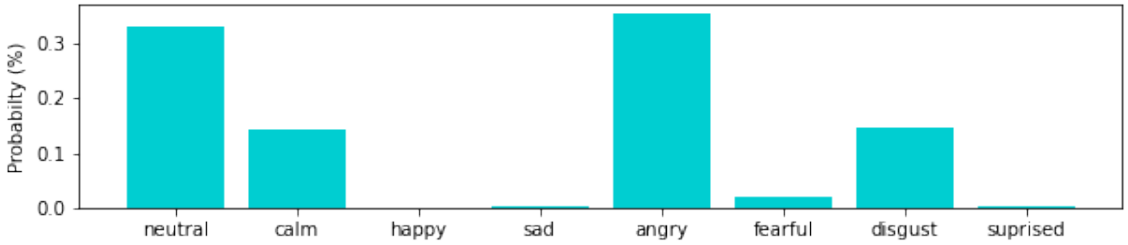

Figure 6.7: Angry Emotion

Finally, after the session ended, we were able to experience every emotion that had been predicted for the entire time, with calm being the emotion that had been maximized. The session summary is depicted in the following graphic.

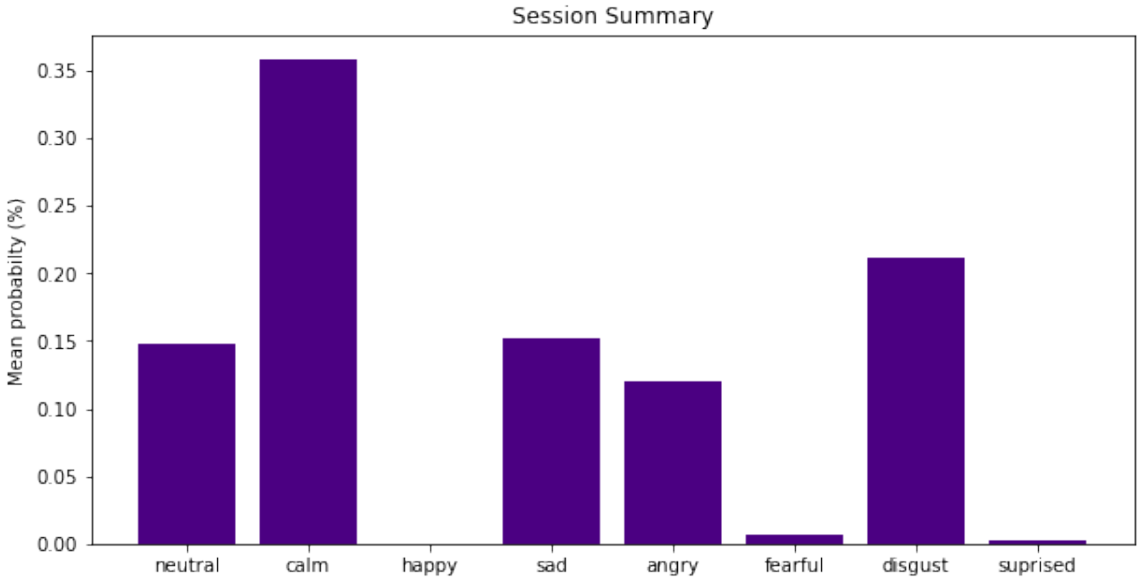

Figure 6.8: Session Summary

As can be observed, we over-fit the training data from the 100th epochs, which resulted in our model over-fitting the validation set accuracy, which was about 80%, and the test set accuracy, which was 85.3%. Numerous models with variable parameters have been attempted, but none of them produced the best results. We establish a model checkpoint where the best weight has been saved while considering the correctness of the model. With that method, we were able to avoid over-fitting.

Chapter 7

Conclusion

A major benefit in the actual world can be gained if the key problem of emotion identification from audio data can be successfully met. Multiple theoretical contributions for future research have been offered in this paper. This study makes an effort to highlight the power of one hot encoding-based feature extraction with padded parameters that is the best fit for an RNN-based LSTM model and complex emotion detection task from speech, even though three feature engineering along with one hot encoding-based embedding has utilized for detecting the emotion from audio. The evaluation of our model shows that it outperforms all other state-of-the-art models discussed in this study's literature review when it comes to audio modalities, with a promising result of 81% weighted accuracy and roughly 80% unweighted accuracy after determining the learning rate after 100 epochs at factor 0.1. Due to the variety of features, we think that using a mix of three modalities can increase accuracy. However, it would impede the development of sectors like healthcare and customer service contact centers that only employ audio as a modality. As the model is trained exclusively using audio-based modalities, it will perform optimally in that situation.

Since we outperformed the top result by about 10% in terms of accuracy, our model can be viewed as having potential for speech emotion identification using audio data since it attained approximately 80% accuracy. An experiment recording the emotions of several parties can be modified for future work since the pathway to identify emotion is scalable. Going beyond the fundamental emotions may also play a significant role in the near future, as it may be used to identify complicated emotions and create a final mixed emotion classifier using the probability values obtained from the RNN model [5]. The theoretical underpinnings of Plumtchik's emotion wheel can be useful in interpreting this sophisticated emotion detection [30]. Another potential area for future research is the enhancement of feature engineering through the use of automatic speech identification processes in conjunction with extracted auditory characteristics, which could lead to the potential use of huge audio data. The CNN-based deep learning algorithms have evolved from one hot encoding to feature extraction that employs sequential information, such as wave2vec 2.0, which has recently entered the field of emotion recognition from human dialogue [7].

When compared to other state-of-the-art models, the performance of the limiting factor used for real-time automatic speaker dualization was somewhat lacking. After reviewing this work, it may be possible to construct a fully automated emotion recog-

nition pipeline from human speech in order to solve the problem. If the complicated model and parameter settings are in harmony, it is anticipated that accuracy will increase. In the absence of reliable real-time approaches, this research suggested the feature pipeline for real-time emotion identification from human speech via manual diary conversations. By integrating the suggested system with a real-time speaker diarising methodology, it is possible to deploy it with significant commercial applications. A method for creating speech embeddings from a large corpus of speech data can be more accurate since the created embeddings reflect rich information while the one-hot encodings have a high dimension and high sparsity. As a useful invention, our work has opened a door for numerous tech-driven real-world industry sectors. If simple and complicated human emotions could be recognized by machines, numerous systems, including virtual call centers, elder care agents, and specially created AI robots, could provide better services. Utilizing this innovation, a company founded on customer contentment might gain more advantages than previously and study how its representatives should handle clients as they exhibit a range of emotions during a conversation. Although a human can interpret complicated emotions through talk fairly easily, machines have a very tough time predicting the correct emotion from just audio dialogue without any facial expression. In this regard, enabling machines to comprehend human emotions through audio dialogues is a huge step forward in the development of human-computer interaction through the improved exploitation of massive audio data.

Bibliography

[1] R. Plutchik, "The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice," *American scientist*, vol. 89, no. 4, pp. 344–350, 2001.

[2] R. W. Picard, "Affective computing: From laughter to ieee," *IEEE Transactions on Affective Computing*, vol. 1, no. 1, pp. 11–17, 2010.

[3] C.-C. Lee, E. Mower, C. Busso, S. Lee, and S. Narayanan, "Emotion recognition using a hierarchical binary decision tree approach," *Speech Communication*, vol. 53, no. 9-10, pp. 1162–1171, 2011.

[4] F. Eyben, F. Weninger, F. Gross, and B. Schuller, "Recent developments in opensmile, the munich open-source multimedia feature extractor," New York, NY, USA: Association for Computing Machinery, 2013, ISBN: 9781450324045. DOI: 10.1145/2502081.2502224. [Online]. Available: https://doi.org/10.1145/2502081.2502224.

[5] C. E. Izard, *Human emotions*. Springer Science & Business Media, 2013.

[6] J. Ruusuvuori, "16 emotion, affect and conversation," *The handbook of conversation analysis*, p. 330, 2013.

[7] K. Han, D. Yu, and I. Tashev, "Speech emotion recognition using deep neural network and extreme learning machine," in *Interspeech 2014*, 2014.

[8] S. A. Alvi, B. Afzal, G. A. Shah, L. Atzori, and W. Mahmood, "Internet of multimedia things: Vision and challenges," *Ad Hoc Networks*, vol. 33, pp. 87–111, 2015. DOI: 10.1016/j.adhoc.2015.04.006.

[9] G. Keren and B. Schuller, "Convolutional rnn: An enhanced model for extracting features from sequential data," in *2016 International Joint Conference on Neural Networks (IJCNN)*, 2016, pp. 3412–3419. DOI: 10.1109/IJCNN.2016.7727636.

[10] M. Schmitt, F. Ringeval, and B. Schuller, "At the border of acoustics and linguistics: Bag-of-audio-words for the recognition of emotions in speech," 2016.

[11] S. Mirsamadi, E. Barsoum, and C. Zhang, "Automatic speech emotion recognition using recurrent neural networks with local attention," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 2227–2231. DOI: 10.1109/ICASSP.2017.7952552.

[12] A. Satt, S. Rozenberg, and R. Hoory, "Efficient Emotion Recognition from Speech Using Deep Learning on Spectrograms," in *Proc. Interspeech 2017*, 2017, pp. 1089–1093. DOI: 10.21437/Interspeech.2017-200.

[13] S. Abeysinghe, I. Manchanayake, C. Samarajeewa, *et al.*, "Enhancing decision making capacity in tourism domain using social media analytics," in *2018 18th International Conference on Advances in ICT for Emerging Regions (ICTer)*, 2018, pp. 369–375. DOI: 10.1109/ICTER.2018.8615462.

[14] D. Hazarika, S. Poria, R. Mihalcea, E. Cambria, and R. Zimmermann, "ICON: Interactive conversational memory network for multimodal emotion detection," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 2594–2604. DOI: 10.18653/v1/D18-1280. [Online]. Available: https://aclanthology.org/D18-1280.

[15] D. Hazarika, S. Poria, A. Zadeh, E. Cambria, L.-P. Morency, and R. Zimmermann, "Conversational memory network for emotion recognition in dyadic dialogue videos," in *Proceedings of the conference. Association for Computational Linguistics. North American Chapter. Meeting*, NIH Public Access, vol. 2018, 2018, p. 2122.

[16] S. Poria, D. Hazarika, N. Majumder, G. Naik, E. Cambria, and R. Mihalcea, "Meld: A multimodal multi-party dataset for emotion recognition in conversations," *arXiv preprint arXiv:1810.02508*, 2018.

[17] S. Yoon, S. Byun, and K. Jung, "Multimodal speech emotion recognition using audio and text," in *2018 IEEE Spoken Language Technology Workshop (SLT)*, 2018, pp. 112–118. DOI: 10.1109/SLT.2018.8639583.

[18] N. Majumder, S. Poria, D. Hazarika, R. Mihalcea, A. Gelbukh, and E. Cambria, "Dialoguernn: An attentive rnn for emotion detection in conversations," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, pp. 6818–6825, Jul. 2019. DOI: 10.1609/aaai.v33i01.33016818. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/4657.

[19] S. Tripathi, A. Kumar, A. Ramesh, C. Singh, and P. Yenigalla, "Deep learning based emotion recognition system using speech features and transcriptions," *arXiv preprint arXiv:1906.05681*, 2019.

[20] D. Alahakoon, R. Nawaratne, Y. Xu, D. De Silva, U. Sivarajah, and B. Gupta, "Self-building artificial intelligence and machine learning to empower big data analytics in smart cities," *Information Systems Frontiers*, pp. 1–20, 2020.

[21] W. Jiao, M. Lyu, and I. King, "Real-time emotion recognition via attention gated hierarchical memory network," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 8002–8009.

[22] I. Madhavi, S. Chamishka, R. Nawaratne, V. Nanayakkara, D. Alahakoon, and D. De Silva, "A deep learning approach for work related stress detection from audio streams in cyber physical environments," in *2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, vol. 1, 2020, pp. 929–936. DOI: 10.1109/ETFA46521.2020.9212098.

[23] A. Adikari and D. Alahakoon, "Understanding citizens' emotional pulse in a smart city using artificial intelligence," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 4, pp. 2743–2751, 2021. DOI: 10.1109/TII.2020.3009277.

[24] A. Adikari, D. Burnett, D. Sedera, D. de Silva, and D. Alahakoon, "Value co-creation for open innovation: An evidence-based study of the data driven paradigm of social media using machine learning.," *International Journal of Information Management Data Insights*, vol. 1, no. 2, p. 100 022, 2021.

[25] A. Adikari, G. Gamage, D. De Silva, N. Mills, S.-M. J. Wong, and D. Alahakoon, "A self structuring artificial intelligence framework for deep emotions modeling and analysis on the social web," *Future Generation Computer Systems*, vol. 116, pp. 302–315, 2021.

[26] A. Adikari, R. Nawaratne, D. De Silva, S. Ranasinghe, O. Alahakoon, D. Alahakoon, *et al.*, "Emotions of covid-19: Content analysis of self-reported information using artificial intelligence," *Journal of medical Internet research*, vol. 23, no. 4, e27341, 2021.

[27] E. Lieskovská, M. Jakubec, R. Jarina, and M. Chmulík, "A review on speech emotion recognition using deep learning and attention mechanism," *Electronics*, vol. 10, no. 10, p. 1163, 2021.

[28] S. Chamishka, I. Madhavi, R. Nawaratne, *et al.*, "A voice-based real-time emotion detection technique using recurrent neural network empowered feature modelling," *Multimedia Tools and Applications*, vol. 81, no. 24, pp. 35 173–35 194, 2022.