Multimodal Emotion Recognition from Speech and Text Using Heterogeneous Ensemble Techniques

by

Sheikh Md Rafidul Islam
22141059
Maria Gomes
22141070
Mehran Hossain
22141043
Ramisha Raihana
21241077

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
May 2022

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Sheikh Md Rafidul Islam
22141059

Maria Gomes
22141070

Mehran Hossain
22141043

Ramisha Raihana
21241077

Approval

The thesis/project titled "Multimodal Emotion Recognition from Speech and Text Using Heterogeneous Ensemble Techniques" submitted by

1. Sheikh Md Rafidul Islam(22141059)
2. Maria Gomes(22141070)
3. Mehran Hossain(22141043)
4. Ramisha Raihana(21241077)

Of Spring, 2022 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on May, 2022.

Examining Committee:

Supervisor:
(Member)

A. M. Esfar-E-Alam
Lecturer
Department of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)

Mobashir Monim
Lecturer
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Emotion recognition and sentiment analysis serves many purposes from analyzing human behavior under specific conditions to enhancement of customer experience for various services. In this paper, a multimodal approach is used to identify 4 classes of emotions by combining both speech and text features to improve classification accuracy. The methodology involves the implementation of several models for both audio and text domains combined using 4 different heterogeneous ensemble techniques - hard voting, soft voting, blending and stacking. The effects of the different ensemble learning methods on the accuracy for the multimodal classification task are also investigated. The results of this study show that stacking is the highest performing ensemble technique, and the implementation outperforms several existing methods for 4-class emotion detection on the IEMOCAP dataset, obtaining a weighted accuracy of 81.2%.

Keywords: multimodal, ensemble learning, emotion recognition, speech, text, stacking, IEMOCAP

Dedication

We dedicate this thesis to our families and friends, who have encouraged us to complete our paper successfully during difficult times, and have supported us from start to finish.

Acknowledgement

We would first like to thank our supervisor A. M. Esfar-E-Alam for providing us the necessary guidance for completing this paper and for his assistance in the research involved.

We would also like to thank our co-supervisor Mobashir Monim for providing insight into the technicalities of writing the paper and for his unrelenting support in answering our queries.

Finally, we would like to thank all the faculties, tutors and members of the Computer Science and Engineering Department of BRAC University, who have helped us further our understanding of the fundamentals involved in our research over 3 years.

Table of Contents

Declaration i

Approval ii

Abstract iii

Dedication iv

Acknowledgment v

Table of Contents vi

List of Figures viii

List of Tables ix

Nomenclature x

1 Introduction 1
- 1.1 Background & Motivation . . . 1
- 1.2 Problem Statement . . . 1
- 1.3 Research Objectives . . . 2
- 1.4 Paper Outline . . . 2

2 Related Work 3
- 2.1 Speech . . . 3
- 2.2 Text . . . 3
- 2.3 Multimodal Classification . . . 4
- 2.4 Ensemble Methods . . . 5

3 Dataset & Preprocessing 6

4 Feature Extraction 8
- 4.1 Speech Features . . . 8
- 4.2 Text Features . . . 14

5 Proposed Methodology 15
- 5.1 Base Models . . . 16
- 5.2 Ensemble Techniques . . . 17

6 Experimental Setup **22**
6.1 Dataset . . . 22
6.2 Model Setup . . . 22
6.3 Hyperparameter Tuning . . . 23
6.3.1 Logistic Regression . . . 23
6.3.2 Random Forest Classifier . . . 23
6.3.3 Support Vector Machine . . . 24
6.3.4 Multinomial Naive Bayes . . . 24
6.3.5 XGBoost . . . 24
6.3.6 MultiLayer Perceptron . . . 24

7 Experimental Results & Discussion **25**

8 Limitations & Improvements **27**

9 Conclusion **28**

References **30**

List of Figures

3.1 Dataset before preprocessing 6
3.2 Dataset after preprocessing 7

4.1 Visualization of MFCC (happy) 8
4.2 Visualization of MFCC (sad) 9
4.3 Visualization of Mel (happy) 9
4.4 Visualization of Mel (sad) 10
4.5 Visualization of Chroma (happy) 10
4.6 Visualization of Chroma (sad) 11
4.7 Visualization of RMSE (happy) 11
4.8 Visualization of RMSE (sad) 12
4.9 Visualization of ZCR (happy) 12
4.10 Visualization of ZCR (sad) 13
4.11 Visualization of Spectral Contrast (happy) 13
4.12 Visualization of Spectral Contrast (sad) 14

5.1 Workflow of proposed model 15
5.2 Stacking Ensemble 18
5.3 Blending Ensemble 19
5.4 Hard Voting Ensemble 20
5.5 Soft Voting Ensemble 21

7.1 Normalized Confusion Matrix showing per-class accuracy of E4 . . . 26

List of Tables

7.1 Performance of each model (WA and F1 scores) 25
7.2 Comparison with state-of-the-art results for classifying four emotions on the IEMOCAP dataset 26

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

$BiLSTM$ Bidirectional Long Short-Term Memory

CNN Convolutional Neural Network

EEG Electroencephalogram

GB Gradient Boosting

$GloVe$ Global Vectors for Word Representation

HCI Human Computer Interaction

$MFCC$ Mel-Frequency Cepstral Coefficients

MLP Multilayer Perceptron

RF Random Forest

RNN Recurrent Neural Network

SMO Sequential Minimal Optimization

SVC Support Vector Classifier

SVM Support Vector Machine

$TF - IDF$ Term Frequency — Inverse Document Frequency

WA Weighted Accuracy

Chapter 1

Introduction

1.1 Background & Motivation

Automated emotion recognition of users by machines could vastly improve user experience when it comes to a variety of services. Emotions play a very significant role in human communication. Analyzing the sentiment of an individual could go a long way in making an impact in numerous fields such as medical areas, call-centers, marketing practices etc. In medical fields, for example, it can be used to judge the mental health of patients. In call centers, it can facilitate automated customer reaction and feedback. Advertisement fields can use such classification to observe how potential customers react to their products. Even criminal investigation departments could utilize such recognition tools to read the psychological conditions of criminals. Though computers may not be able to mimic this recognition ability completely given the complex nature of emotions, its detection can aid in improving the human computer interaction (HCI) field. By incorporating principal emotions of a speaker into speech processing, machines can act more naturally and adapt to treat specific users differently. Emotions can be classified based on facial cues, gestures, speech, text or even EEG signals from the brain. However, due to the limited access to human features most computers may have in real life scenarios, we limit our studies to speech and text as they are more plausible compared to facial expressions or even brain signals.

1.2 Problem Statement

Previous works such as [22] and [20] have implemented neural network based models to recognize emotions from a speech, while [7], [11] and [13] have tried using lexical features to achieve the same results. These approaches are unimodal as they focus solely on a single kind of channel of expression. However, this overlooks the contribution of these features if used in conjunction with each other. To elaborate, the sentiments found from text can be interpreted in different ways depending on the tone and vocal features. For instance, the text "How could you do that?" can portray different feelings depending on the tone of speech. It can convey anger if said in a loud and aggressive manner, or express sadness if a slow and quiet tone is used. Likewise, the same can be said for speech, where similar speech features can be found in different emotional expressions and can only be differentiated by investigating the spoken content i.e the lexical data. This is because people use cer-

tain words depending on their feelings as they associate specific words with certain emotions, such as using swear words when they're angry. So, it can be said that one domain provides additional but essential contextual information paired with the other. Linguistic information integrated with acoustic features can give a richer set of intention-focused characteristics used to train a classification model.

Therefore, we can conclude that relying solely on either one feature would not suffice to give us an accurate emotion classification. Choosing a unimodal approach in such a case could result in a loss of context needed for classification. To solve this problem, we use the correlation between text and speech to pursue a bimodal approach which seeks to use both sets of features to train classifiers separately and then combine them using a number of ensemble techniques, then choosing the best one for a combined prediction.

While lexical features focus on words, phrases, symbols and sentences, speech focuses on different spectral and prosodic features such as pitch, energy, MFCCs etc. The novelty of our work lies within the way we joined the two modalities. To combine them, we implemented 4 different ensemble techniques which can combine results from different models. Among them, voting techniques like hard and soft voting have been tried before in previous approaches like in [19] and [14]. However, two-layer ensemble methods such as Stacking and Blending have not been suggested in other works for this task to the best of our knowledge.

1.3 Research Objectives

Considering the challenges in existing methods and the motivation behind this study, our research objectives are:

- Double layered ensemble techniques such as Stacking and Blending used to combine modalities beside old Voting methods
- Comparing performance of different heterogeneous ensemble techniques
- Merging acoustic and linguistic information to recognize emotions of a speaker

1.4 Paper Outline

From here on out, our paper is structured as follows: Section 2 shows the research we did into existing methods for emotion recognition. We divide this section into four parts, where we review work done using speech, text, combination of the previous two, and previous approaches utilizing ensemble techniques. Section 3 talks about the dataset we chose for our task and how we processed it for feature extraction. Section 4 is about the features we chose from our data, divided into two parts for each modality. Section 5 dives into the approach we took and details the different ensemble techniques we used. Our entire process and setup is described in Section 6, the results of which are then shown and compared in Section 7. Lastly, Section 8 talks about the limitations we faced and some possible improvements we could make in the future while Section 9 concludes our work.

Chapter 2

Related Work

2.1 Speech

To detect emotions from speech, authors in [22] have tried a CNN and BiLSTM based model by using a key sequence segment selection, where a spectrogram is produced from it and passed into the CNN for feature extraction. Normalized CNN features are fed to the BiLSTM which learns the temporal features for the final classification. To reduce computing costs, only the key segments are processed by the model. Emotion recognition from speech using MultiLayer Perceptron (MLP) networks has also been explored in a paper [20] that used speech features such as MFCCs, Contrast, Mel Spectrogram Frequency, Chroma and Tonnetz as input from speech data to classify 8 different emotions.The authors implemented three hidden layers in the MLP model and achieved a 70% accuracy on the Ravdess dataset using logistic functions instead of ReLu activation functions. Their studies showed that selecting appropriate speech features such as MFCCs had a significant impact on the performance of the model.

2.2 Text

LSTM-based methods have been the focus of many recent studies on emotion detection in text. One study [7] extracted and combined different features from text data such as semantic word vectors obtained using Word2Vec and passed them through the LSTM model. An accuracy of 70.66% was found which was a 5.33% improvement over the CNN-based method. Another study [11] implemented a model combining BiLSTM and CNN subnetworks and tested it on text representations using Word2Vec, GloVe, and FastText. The best results were found using FastText. The study also compared the model to deep learning and traditional machine learning models. The results showed deep learning models performed better overall while the proposed model outperformed the other models in nine out of ten datasets used.

SVM, LSTM and Nested LSTM were used for sentiment analysis from text in [13]. After utilizing the Twitter API, they successfully retrieved text by tweet id which resulted in 980,549 training and 144,160 testing data. TF-IDF feature extraction method was used in this paper and after all the research, they found out that the Nested LSTM method performed the best with an accuracy score of 99.167% which was not significantly different from the result of LSTM (99.154% accuracy).

However, LSTM had a better average score for recall, precision and f1 score which were 98.86%, 99.22%,and 99.04% respectively.

2.3 Multimodal Classification

A combination of speech and lexical features have been proposed before by different scholars. G. Sahu achieved impressive results in [14] by soft voting twelve classifiers, six for each mode. Using prosodic features such as pitch, harmonics, speech energy etc. on the IEMOCAP dataset [2] and the text transcriptions together, the author classified six emotions with an approximated 14% higher accuracy compared to either modes alone. Similarly, the multimodality of the text-audio classification task has been explored in a study [16] where an attention mechanism was implemented to create multimodal alignment features using text and speech data rather than creating separate models for each domain and combining their outputs. The study involved the use of a speech encoder to extract low level speech features from the audio data and a text encoder to obtain text features, both consisting of a bidirectional LSTM. Once the encoders have extracted the speech and text features, they are combined together in the attention layer. The LSTM and attention based model outperformed existing methods and could be improved by using better speech recognition techniques.

Another study used dual recurrent neural networks (RNNs) to process speech and text data simultaneously [9]. The audio encoding vector and textual encoding vector obtained from the RNNs were concatenated and passed through a function to determine the emotion. The model achieved accuracies of 68.8% to 71.8% on four emotions on the IEMOCAP dataset and outperformed many earlier models.

Researchers in [8] first found the best architecture for each model classification and only performed fusion at the final layer which gave the advantage of using a modular approach for faster interference time and training. They also pointed out that it was easier for them to replace any model with a better one because of this approach. Their ensemble consisted of CNN's, Long Short Term Memory Networks, Multi-Layer Perceptions that were fully connected and all these were complemented with techniques like Attention based RNN decoders, Dropout, Adam (an adaptive optimizer) and pre-trained word embedding models. For the final model, they chose the best text (stacked LSTMs which used Glove word embedding) and speech model (bidirectional LSTMs with attention which were 2 stacked and it was also combined with Mocap Model1 which had stacked convolutional layers) from their experiments. Feature fusion was performed after that and finally, another layer was added which was fully connected (with 256 neurons).

Combined CNN models for text and speech were used in [15]. The authors compared models using spectrograms and MFCC, spectrograms and text data, and text data and MFCC. They found that models using text data scored better, with the text-MFCC model having the highest unweighted accuracy at 76.1%.on IEMOCAP. Another study [12] combined CNN and LSTM models for handcrafted acoustic features, while a Bi-LSTM model was used for text features. Finally, text and acoustic features were fused to get high level features on which a deep neural network was

trained for classification. The authors found the multimodal model to perform better than base models, showing the effectiveness of feature fusion.

2.4 Ensemble Methods

Ensemble approaches on unimodal detection of emotions have been tried before. For example, [19] tried an ensemble of Random Forest (RF), Gradient Boosting (GB) and Hist GB by using a voting classifier. By using spectral features of speech such as MFCC, Mel, Chroma etc. they first tested six different classifiers individually. The best three models were then chosen to be ensembled. Although it is unclear whether soft or hard voting was implemented, they found that the voting classifier performed 10 to 13% better than the individual models to classify eight emotions on the RAVDESS dataset [5]. Another study [6] showed that ensemble learning using several models performed better in speech emotion recognition tasks than the individual models themselves. The models used included Arousal-Valence models and typical categorical learning models. The authors implemented 3 types of ensemble classifiers and the classifier with combined input vectors from categorical, arousal and valence models performed the best, with an accuracy of 82.1%.

One paper [24] used a majority voting technique of ensemble learning on cross-corpus multilingual data for speech emotion recognition. They used four Corpora(EMO-DB, URDU, SAVEE, and EMOVO) to maintain the language diversity. In their research, they noticed that a single classifier wasn't fit to choose as (RF, SMO, and J48) they were'nt giving the best results individually for all the datasets. In the end, they decided to combine the effect of all the classifiers for the cross-corpus model as this ensemble gave the best results.

Another study [4] used ensemble learning to deal with the problem of skewed data in datasets. Different selection methods were used on the highly populated neutral class of the FAU-Aibo dataset and used to train an ensemble of neural network based classifiers. The highest unweighted average (UA) was obtained when the outputs of the ensemble were used as features and fed into a neural network.

In [18], weighted voting is used in an ensemble system of three deep learning models focusing on different aspects of emotion recognition using local features of spectograms, local statistical features, etc. The authors found that two models were capable of correcting misclassifications of the third model to achieve better scores. Data imbalances had a smaller effect on the ensemble model as confirmed by its higher WA score (75% on IEMOCAP). Thus, combining models specializing in different areas can be an effective technique.

Chapter 3

Dataset & Preprocessing

The IEMOCAP [2] dataset was used for the multimodal classification task as it consisted of 9 emotions and around 10k audio recordings, both scripted and improvised, with transcripts across several sessions. This presence of both audio and its corresponding textual data in the dataset makes it a good fit for our multimodal approach. Due to the imbalanced nature of the dataset, sparse emotions such as disappointment, surprise, frustration and fear were dropped along with ambiguous categories such as 'xxx' and 'oth'. Excited and happy labels were merged to happy as well due to the closeness of the categories. Hence the implementation involves 4 major emotions- anger, neutral, happiness and sadness. This was done to not only make the dataset balanced but also to allow comparison with existing studies involving 4-class emotion recognition. The initial preprocessing resulted in 5531 samples and the raw audio data was used for speech feature extraction. The text data for each audio was processed further by removing stop words and punctuation while also lowercasing characters. A few symbols that were indicative of emotions such as '?' and '!' were not removed.

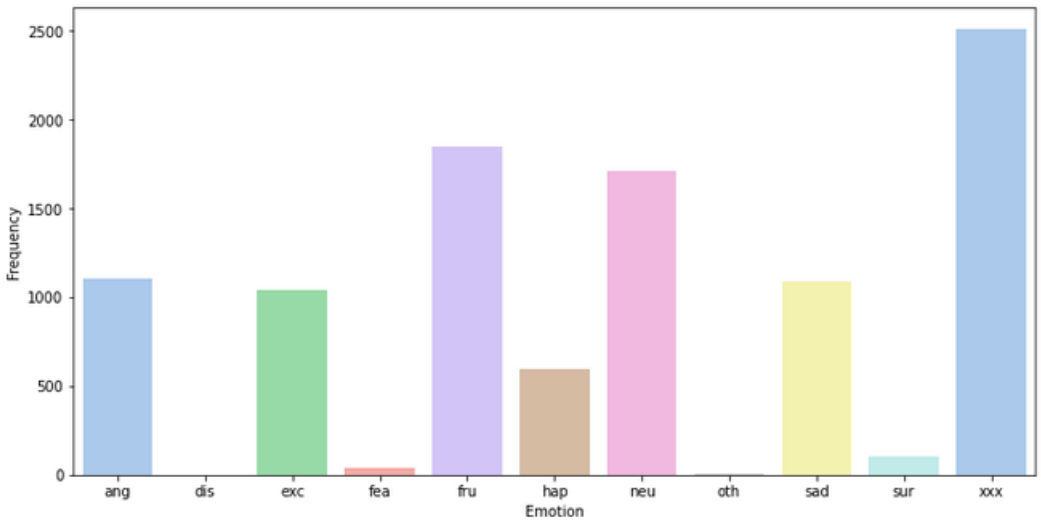

Figure 3.1: Dataset before preprocessing

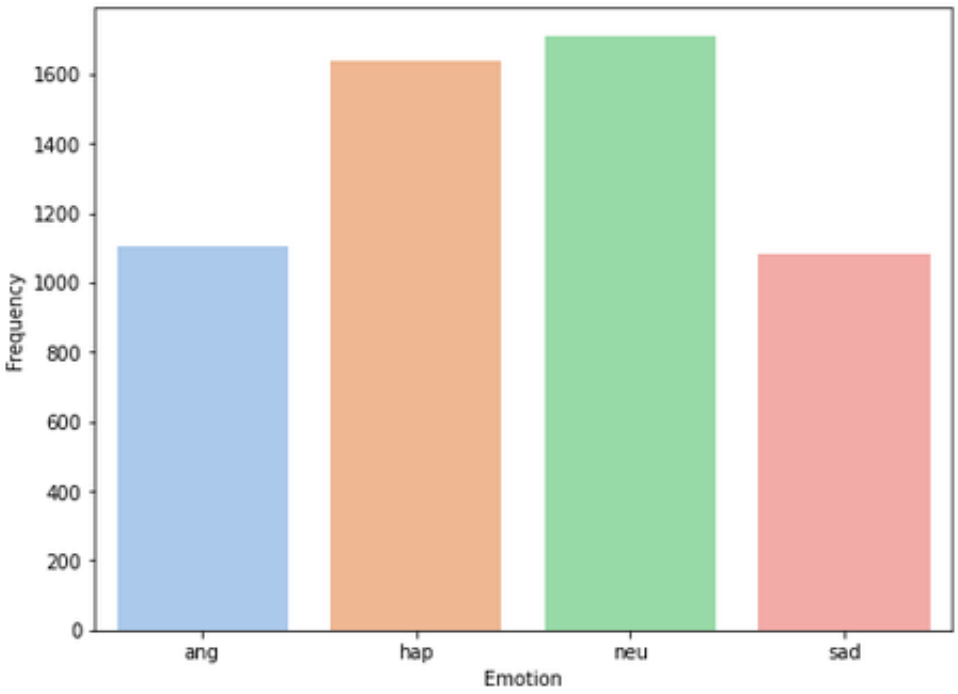

Figure 3.2: Dataset after preprocessing

Chapter 4

Feature Extraction

4.1 Speech Features

To extract meaningful information from the audio files, we mostly used spectral features apart from pitch, which is a prosodic feature. The conversations available in the dataset are broken down into sentences, with each sentence having its corresponding labeled file, from which the features are extracted. Features extracted from the vocal tract system are called spectral features, which are on the frequency domain. They can give us information regarding the movement of articulators and the nature of the vocal tract. Prosodic features on the other hand are concerned with rhythm, stress, intonation etc. They can extract emotional expression or excited behaviors.

For our task, we focus on the following features, a few of which have been visualized:

- MFCC: Mel-Frequency Cepstral Coefficients provide information regarding the shape of the speech signal spectrum. After windowing the speech signal, discrete Fourier transform is applied. The log of the magnitudes are taken and the frequencies are warped on the Mel scale, after which an inverse discrete cosine transform is applied. The first 40 MFCCs are taken for our research.

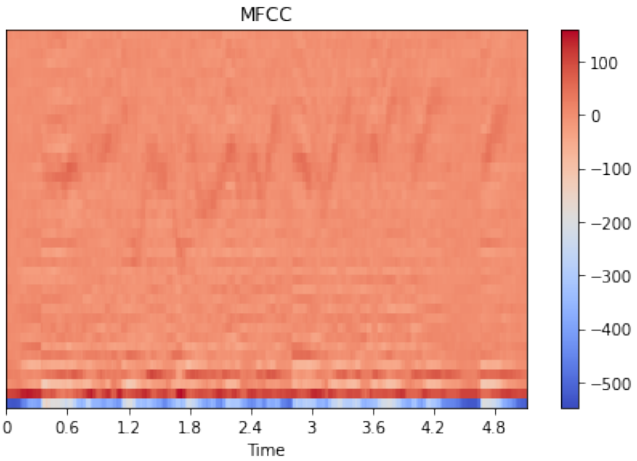

Figure 4.1: Visualization of MFCC (happy)

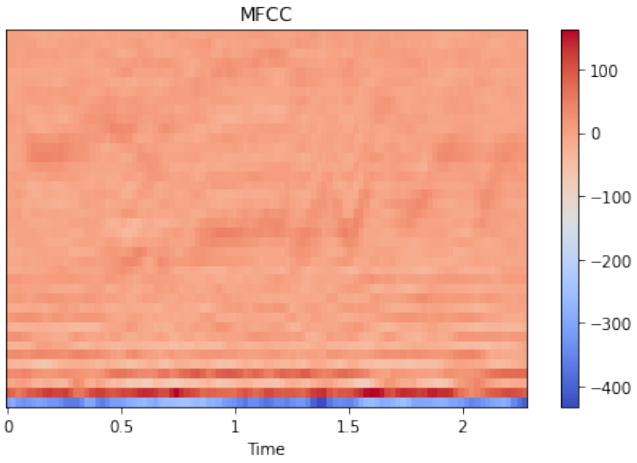

Figure 4.2: Visualization of MFCC (sad)

- Mel-Filter Banks: They mimic the nonlinear frequency feature of human ears. This feature is the ability of humans to differentiate between lower frequencies more easily than higher ones. The Mel scale thus makes it easier to identify differences in lower frequencies with the help of filter banks that separate the input signal into multiple components.

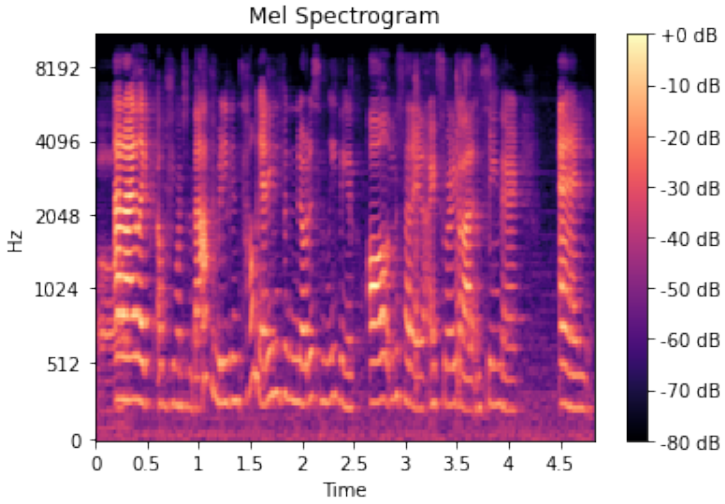

Figure 4.3: Visualization of Mel (happy)

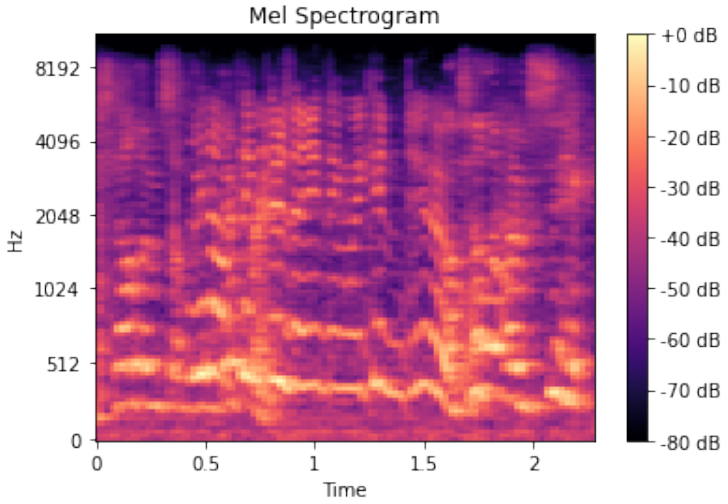

Figure 4.4: Visualization of Mel (sad)

- Chroma: Represents the pitches from the 12 different pitch classes in an audio, giving us the tonal content of the signal.

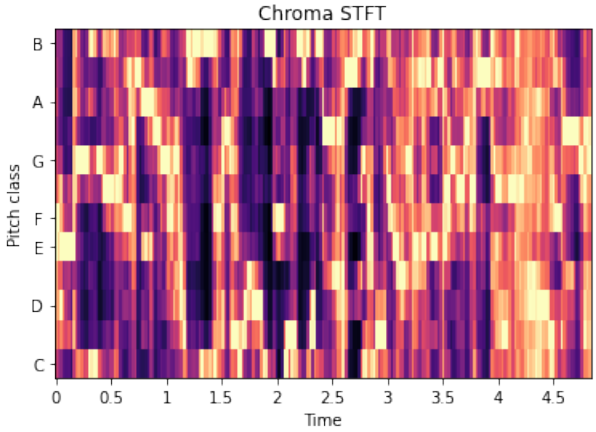

Figure 4.5: Visualization of Chroma (happy)

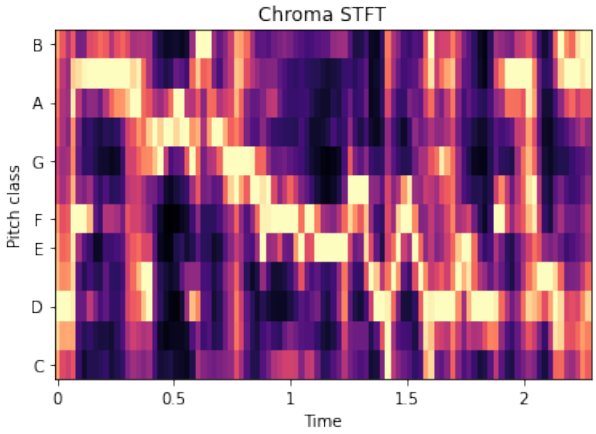

Figure 4.6: Visualization of Chroma (sad)

- Root Mean Square Energy: It is the overall energy of the speech signal, which can also denote loudness. It can be a good indicator for angry or sad emotions.

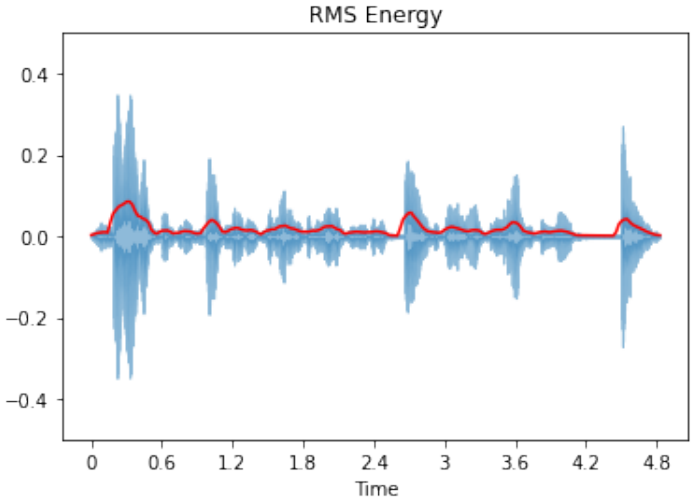

Figure 4.7: Visualization of RMSE (happy)

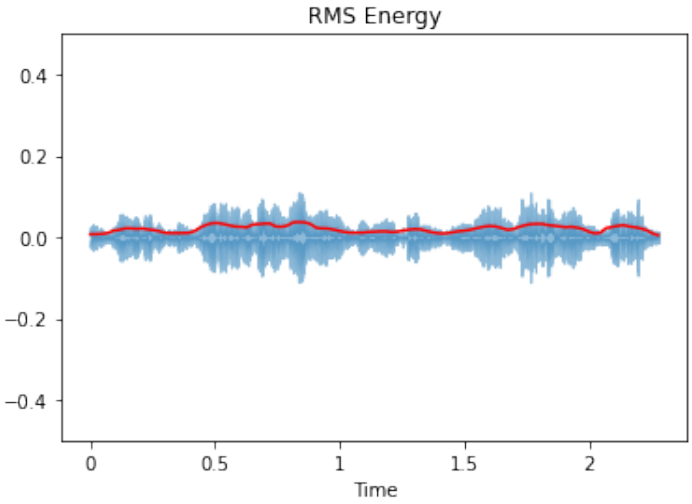

Figure 4.8: Visualization of RMSE (sad)

- Zero Crossing Rate: It is the number of times the signal changes its polarity.

Figure 4.9: Visualization of ZCR (happy)

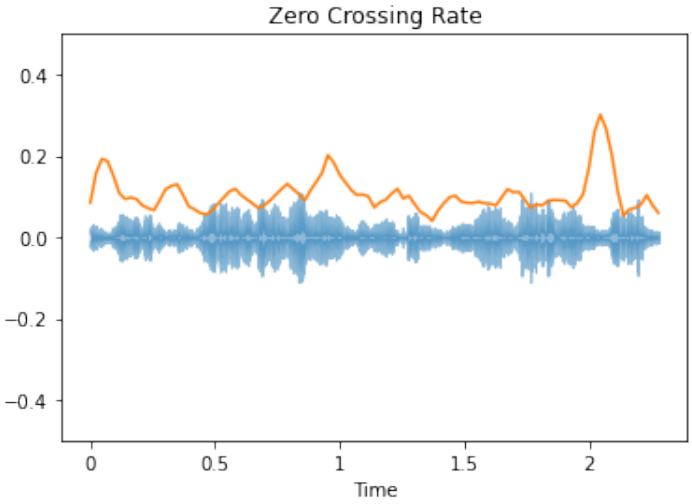

Figure 4.10: Visualization of ZCR (sad)

- Spectral Flux: For an audio signal, this is the spectral change between two consecutive frames. It is calculated by finding the difference between the spectral magnitude of two successive windows and squaring that value.
- Spectral Roll-Off: It is the fraction of bins in the power spectrum at which 85% of the power is at low frequencies. It gives us an idea of high frequency in a signal and gives the frequency where a certain amount of energy can be found.
- Pitch: A measure of the frequency of sound.
- Contrast: It is defined as the difference in decibels between spectral peaks and spectral valleys in a speech signal.

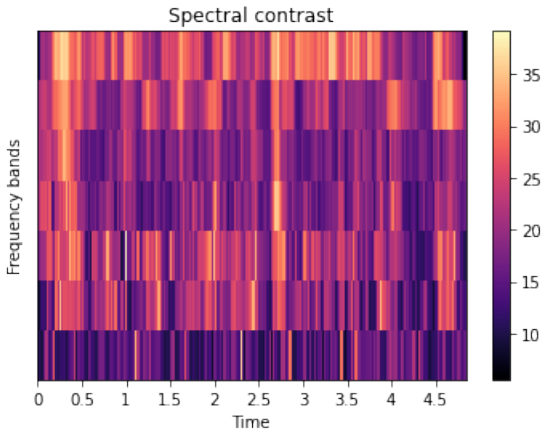

Figure 4.11: Visualization of Spectral Contrast (happy)

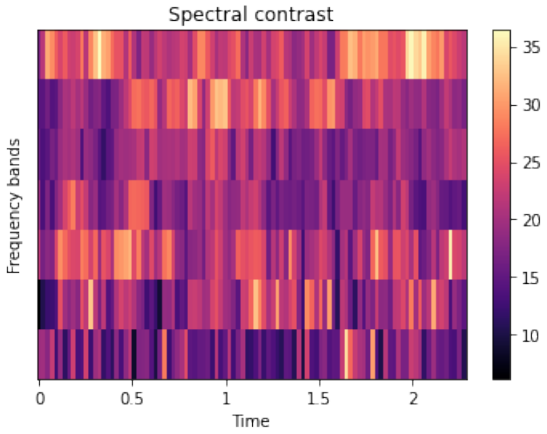

Figure 4.12: Visualization of Spectral Contrast (sad)

4.2 Text Features

Feature extraction or vectorization is the process of encoding the raw text data into floating-point or integer values to use as inputs in the machine learning algorithms. Among many text feature extraction techniques such as Bag of Words, TF-IDF and Word2Vec, the Count Vectorization method was chosen to transform the text data into vectors on the basis of word frequency only. After analyzing both TF-IDF and Count Vectorization, we came to the conclusion that TF-IDF decreases the accuracy because of the class imbalance and limited text corpus to learn from in the dataset. Hence, CountVectorizer was used to convert each word into a vector from each text for further text analysis.

After the initial pre-processing stage, the CountVectorizer method was applied with a few specific parameter values, such as the 'min_df' value set to 5 and 'ngram_range' to (1,2). As the 'min_df' parameter value was 5, words lower than this document frequency count were ignored while building the vocabulary. Moreover, a combination of words holds different meanings, that's why the (1,2) value for the 'ngram_range' parameter was used to consider both bigrams (two-word combinations) and unigrams (single words) for the feature extraction process.

Chapter 5

Proposed Methodology

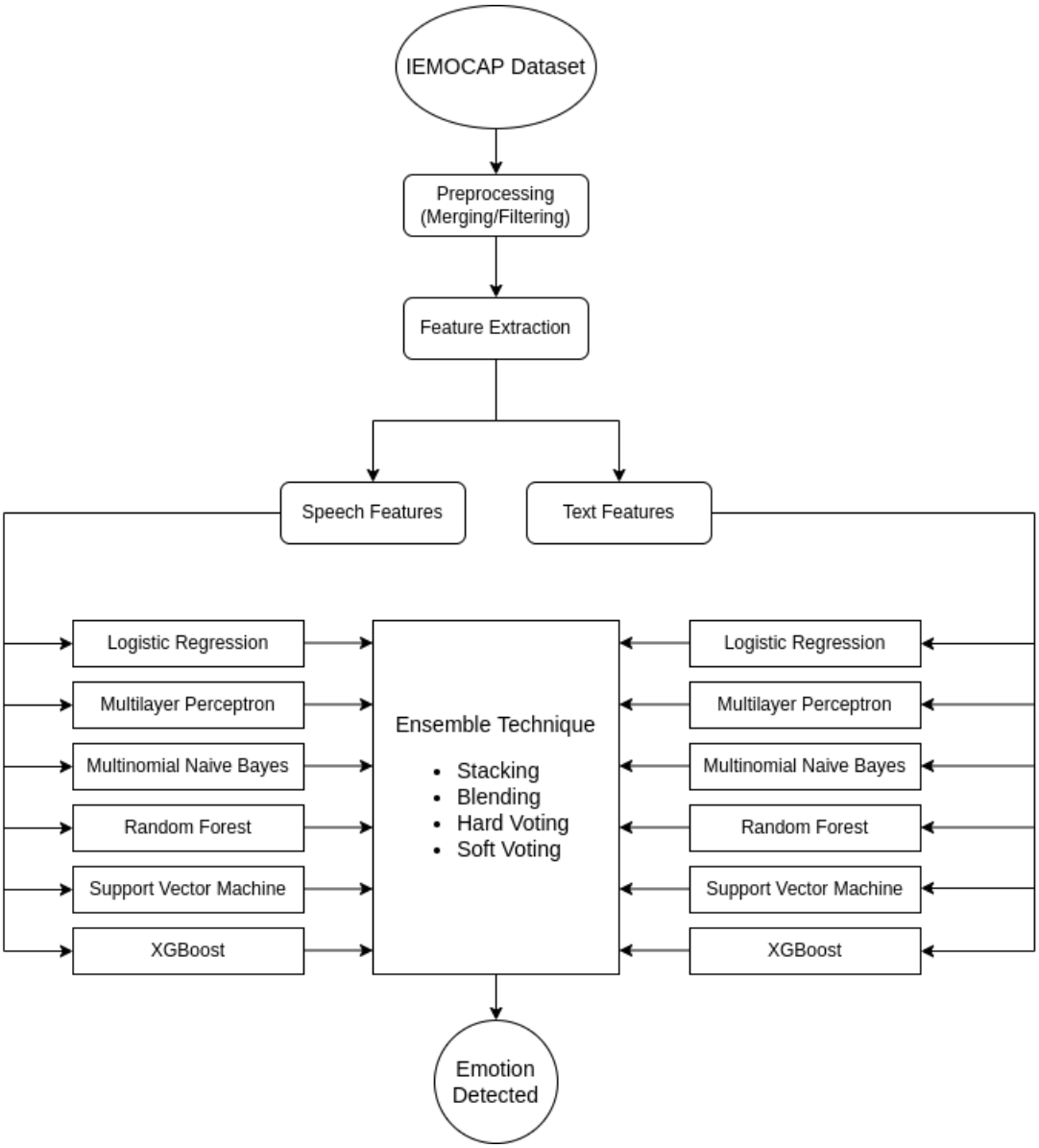

Figure 5.1: Workflow of proposed model

We have taken a multimodal approach using six popular types of base classifiers that learn from features in the speech and text domain. As stated in Section 1, emotional information in text features provide additional context with acoustic features that is necessary to identify the correct emotion. In the real world, emotions are conveyed through a mix of different channels of physical expressions. Emotional information found in either acoustic or linguistic features alone are limited and don't always paint the full picture. Moreover, it is evident from similar approaches such as [14], [8], [15], [10] and many more that speech features complement the lexical data and lead to a more accurate classification. In this study, each base classifier is trained on the two modalities separately, giving us a total of twelve trained models. They are then combined on a decision level using a heterogeneous ensemble technique, which gives us the final prediction.

5.1 Base Models

The base classifiers used are Logistic Regression, Random Forest, Support Vector Machine, Naive Bayes, XGBoost and a neural network model, the Multi-layer Perceptron.

Logistic Regression is one of the most frequently used machine learning models for the prediction of categorical variables. Given that this study aims to predict categories of emotions using speech and text features, the logistic regression model is suitable for the classification task. The Logistic Regression model makes the assumption that the independent variables of the data are not correlated with each other and uses a sigmoidal loss function to output the predicted probabilities to get values between 0 and 1 given a certain threshold value. In this study, a multinomial Logistic Regression model is used as there are several labels that must be predicted which are not ordinal in nature.

The Random Forest classifier is itself a homogenous ensemble of Decision Trees which are constructed based on certain criteria. A commonly used criterion is the GINI impurity between variables, which is the probability that a random sample would be incorrectly labeled depending on the distribution of the samples. These samples can be randomly selected from the dataset using the bootstrapping technique. In this case, the Decision Trees are built in a way that minimizes the GINI impurity. Alternatively, the entropy between randomly selected variables in the sample can also be used as the criteria for the construction of the Decision Trees. Once the desired number of Decision Trees or estimators have been built, they are assembled together using bagging and then used to output a prediction using a selected method such as majority voting.

Support Vector Machines are classification models that aim to create a hyperplane to categorize variables on either side of the plane. In order to create this decision boundary or hyperplane,, the extreme points, or the 'support vectors' in the training dataset are taken. The Support Vector Classifier in particular, is used for this paper, which is a clustering algorithm that does not make assumptions about the number of clusters in the data. It can be used for binary as well as multi class classification. Because of their computationally expensive nature, SVC tends to work better on

low dimensional data, otherwise additional preprocessing methods are required to reduce the number of clusters.

The Naive Bayes classifier is a popular machine learning model for tasks such as sentiment analysis and document classification, because of its simplicity and the fact that it does not require large amounts of data, be it discrete or continuous. It involves the use of the Bayes rule, prior probabilities and conditional probabilities of pairs of features to calculate the output probabilities for each label. It is naive in the sense that it treats all features as independent of other ones. Each feature is also assumed to have an equal effect on the outcome, with none being irrelevant. This paper implements a Multinomial Naive Bayes classifier, which is suitable for discrete features such as word frequencies for the text classification task.

XGBoost is a model that implements extreme gradient boosted decision trees and differs from Random Forest classifiers in the sense that XGBoost deals with the functional space while Random Forest optimizes via hyperparameters. Random samples from the data are taken with replacement to create the dataset on which it tries to determine the best estimator or decision tree. The gradient boosting algorithm builds several decision trees sequentially where the current tree is built based on the errors of the previous ones, thus aiming to create estimators that improve in performance at each step using gradient descent. The loss function differs based on the problem- mean squared error is usually used for regression tasks while log likelihoods are used for classification tasks. In XGBoost, the gradient boosting method is advanced with additional regularization factors and penalties that both reduce training time and create better performing models.

The Multi Layer Perceptron is a feed forward neural network that is fully connected with one or more hidden layers. It is an improvement on simple perceptron networks as it can distinguish data that is not linearly separable. Each neuron can use any arbitrary activation function and the backpropagation algorithm is used to adjust and learn the weights for each neuron. The backpropagation algorithm is optimized using gradient descent, which aims to find the local minimum of a given function. To do this, the gradient of the mean squared error is calculated and the weights are adjusted in a backward pass until the convergence threshold is reached.

5.2 Ensemble Techniques

Ensemble techniques aim at improving the accuracy of results in models by combining multiple models. They are ideal for regression and classification, where they reduce bias and variance to boost the accuracy of models. Rather than depending on a single model for the best solution, ensemble learning utilizes the advantages of several different methods to counteract each model's individual weaknesses. Ensemble techniques can be homogeneous or heterogeneous. The former combines multiple instances of the same kind of model, such as bagging or boosting. However, as the purpose of our research is to combine different models, we try the following heterogeneous ensemble techniques and choose the best performing one as our final model:

- Stacking: First talked about in [1], this method has a series of base classifiers on level-0 and a meta classifier on level-1 which performs the job of combining the output of the base models. The base models are trained on the training data and their predicted probabilities are used as inputs for training the meta model. A k-fold like approach is taken here to cover the whole training set without overfitting. At first, the training data is split into k parts. Then, every base model is trained k times on (k-1) different splits, each time predicting and finding probabilities on the test split. Now, the 'k' number of predictions are joined up and fed as training data along with the true labels to the meta model. Lastly, each base classifier is trained a second time on the whole training set.

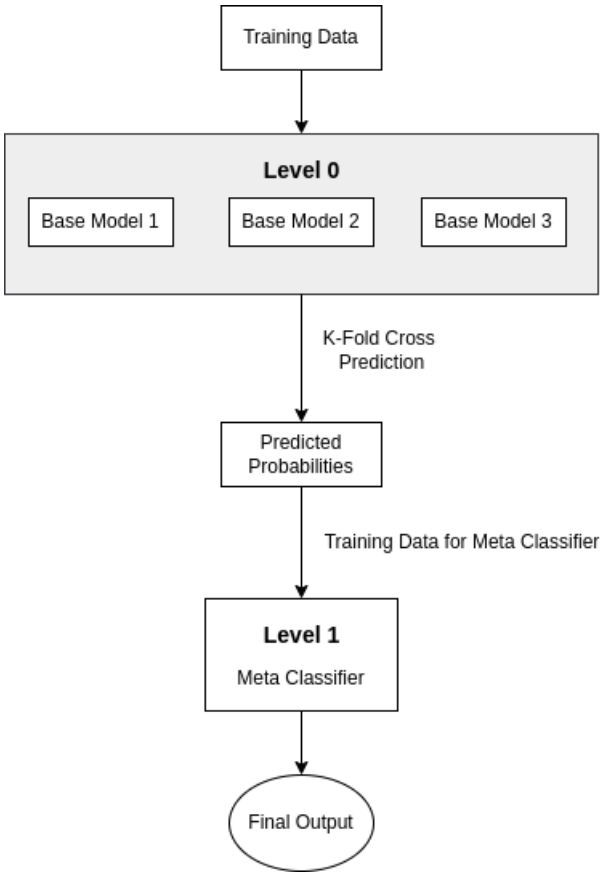

Figure 5.2: Stacking Ensemble

- Blending: It was introduced as a modified stacking model which won the Netflix Grand Prize in 2009 [3]. It is similar to stacking, but instead of a k-fold approach, a portion of the training data is held out as a validation set. Then, each base model is trained on the training set and predicted on the validation set. The predicted probabilities are fed to the meta classifier as training data. (For stacking and blending, we chose to use XGBoost as our meta classifier as it gave the best performance as a meta model).

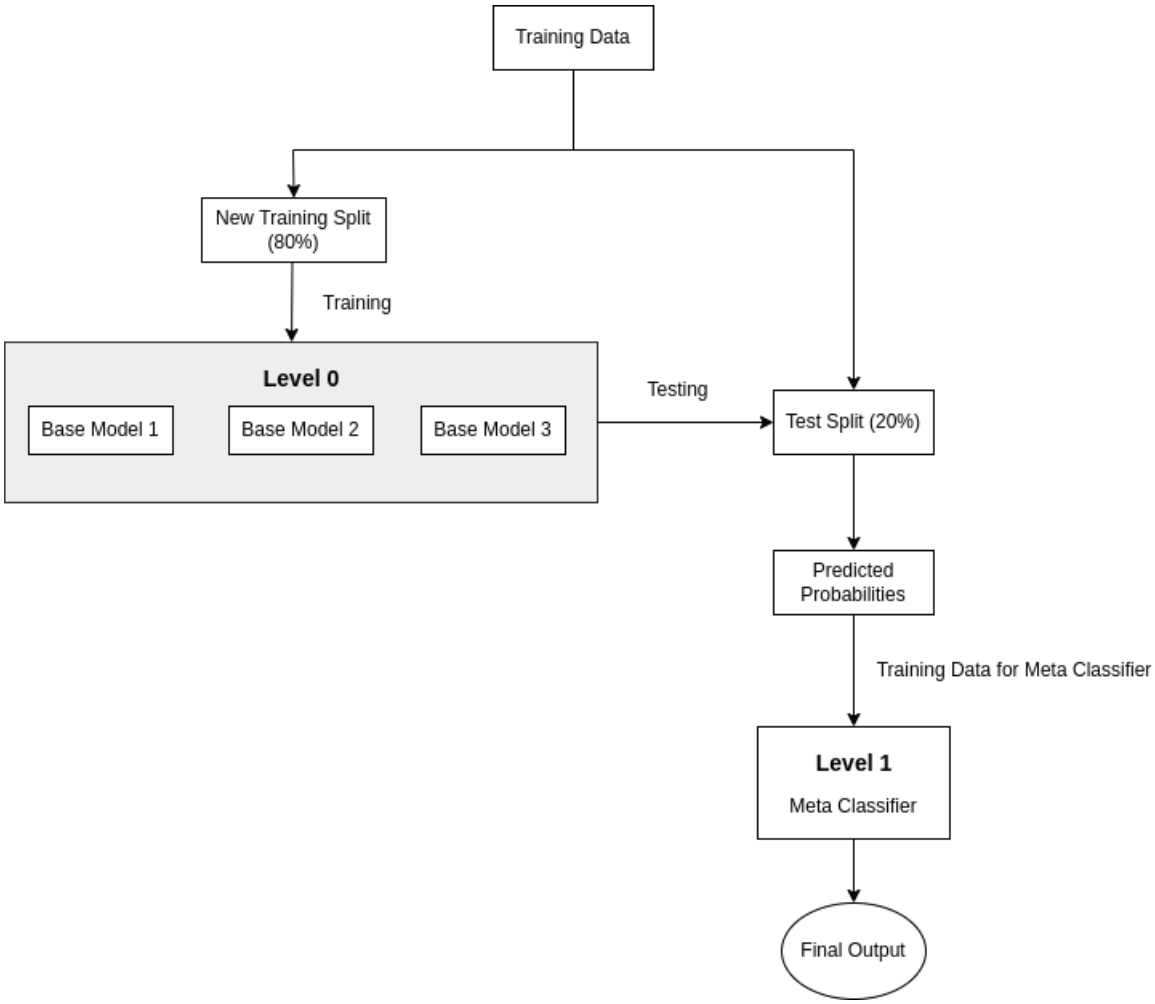

Figure 5.3: Blending Ensemble

- Hard Voting: By far the simplest method of combining models, predictions are done on all base models and the emotion that was chosen by the most models is the final output i.e it uses majority voting.

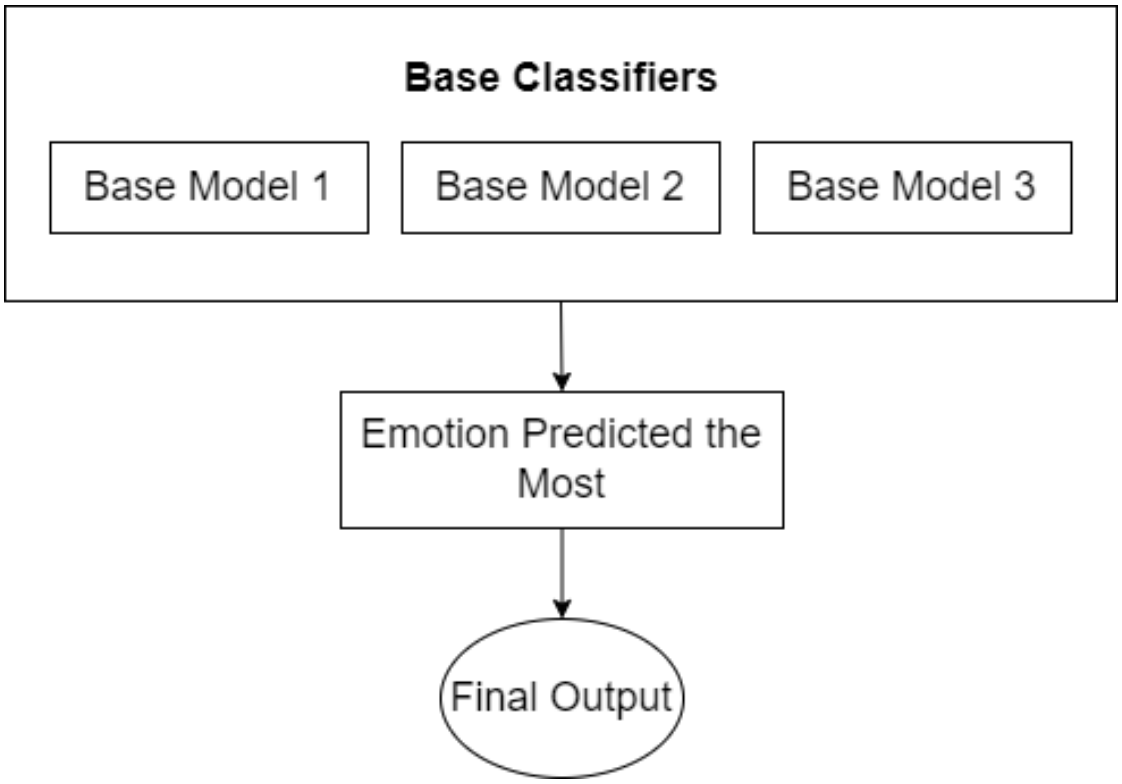

Figure 5.4: Hard Voting Ensemble

- Soft Voting: Here, all the probabilities obtained from each base model when predicting are averaged and the emotion with the highest probability afterwards is the final prediction.

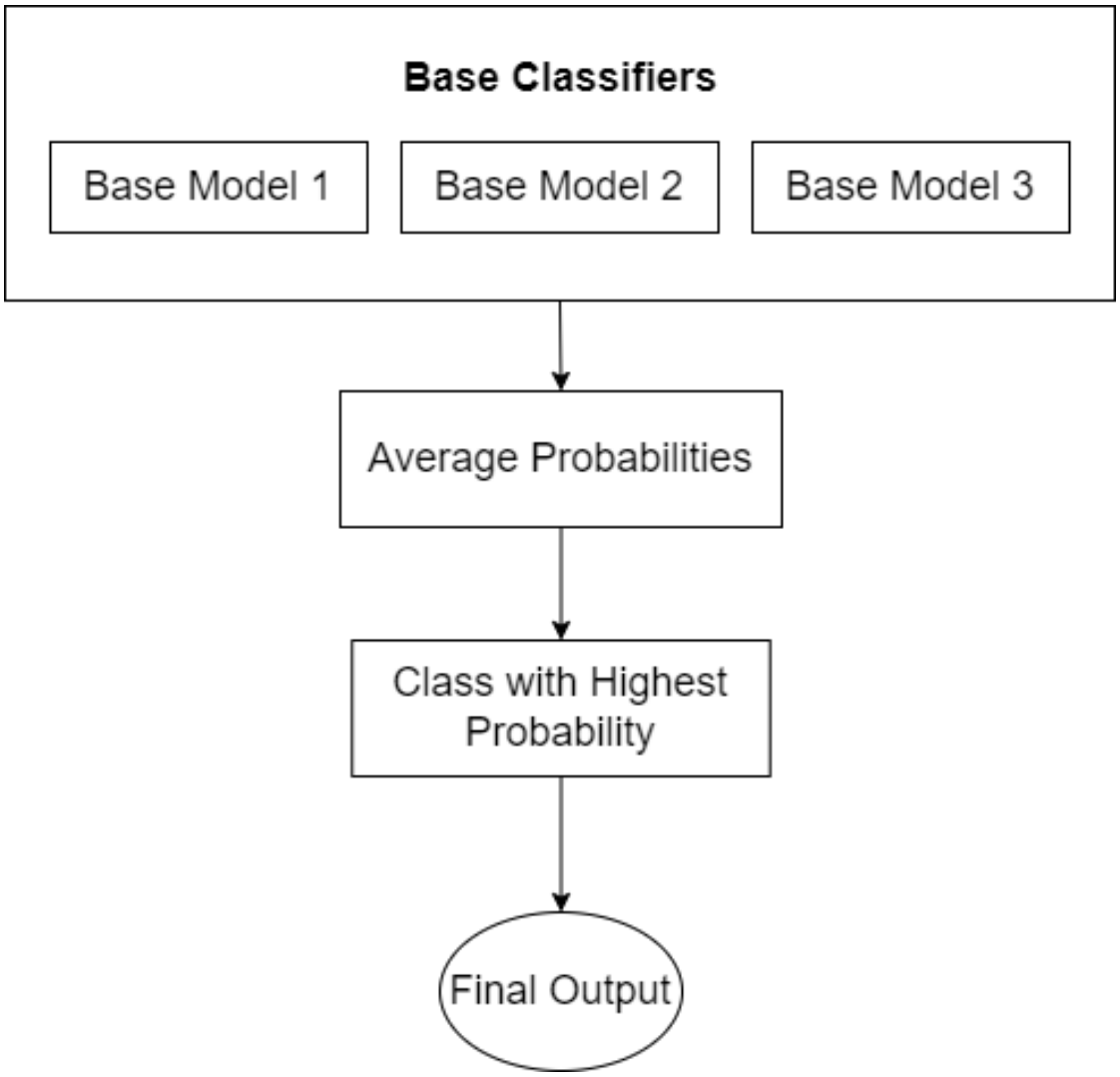

Figure 5.5: Soft Voting Ensemble

Chapter 6

Experimental Setup

6.1 Dataset

We carried out our experiments in Python using the scikit-learn library for our machine learning models. After getting access to the IEMOCAP dataset, we first moved on to processing it for feature extraction. The dataset has recordings of both scripted and improvised conversations in 5 different sessions. For every session, it also breaks down each sentence of every conversation into its own labeled .wav file, which is what we focus on. With a dataframe giving us the file path to every .wav file, we clean the dataset as detailed in Section 3 by merging and removing emotions. Then, we shuffle the file list for a more equal distribution of data as they were originally sorted by speaker name and session.

First, we go through each file and perform speech feature extraction on them. To deal with the audio files, the librosa library was used. The extracted speech feature vectors are then stored in a file using the pickle library to be used by our models. To scale the speech features, a MinMax scaler was used. To find text features, we had to go over the transcript files and find the dialogues with their labeled emotions. A CSV file was made with every dialogue beside its label, after which the sentences are cleaned accordingly. It is then shuffled with the same random seed as the speech dataframe so the dialogues correspond to their respective speech features. A Countvectorizer is then used to vectorize from the CSV files.

6.2 Model Setup

All 12 classifiers are instantiated from the scikit-learn library, which are then imported into the ensemble. Among our ensemble techniques, while Blending techniques cannot be found in any library, Voting Classifiers and Stacking Classifiers can be found in the scikit-learn and mlxtend libraries. However, they cannot be implemented for our purpose. This is because of the multimodal approach we took, which works with two kinds of data.

The classifiers in these libraries can be fitted and predicted on only one kind of domain. As we're not carrying out a feature level fusion which would combine the acoustic features with the lexical features and use them for training, we had to implement the four ensemble methods from scratch in Python. These models were

developed to be trained using both kinds of features by storing the list of speech and text models separately. These models are then trained by using their respective features only. To evaluate our models, we then used k-fold cross validation and averaged the metrics across k samples.

6.3 Hyperparameter Tuning

In order to build the best performing base models in a reliable way, trial and error was used to perform hyperparameter tuning along with k-fold cross validation to eliminate the possibility of overfitting to a specific set of data. In this study, a wide range of parameters were first selected to form a parameter dictionary for each base model. The next step was to instantiate several instances of each base model and fit them to the training data using random combinations of the selected parameters. The models were then tested using 5-fold cross validation to get an overall accuracy for each model. All combinations were not used in the first step as it is computationally expensive. Hence a random grid of accuracies and parameters were created first using the best estimators in the grid, which was then used to narrow down the first set of parameters. Once the number of parameters to test was sufficiently small, all possible combinations of parameters were tested in the same way to find the best ones.

6.3.1 Logistic Regression

The logistic regression model trained on speech data was tuned on a small set of parameters consisting of the type of solver, penalty and C value. The solvers adjust weights to minimize the cost function, the penalty is imposed when regularization is required and the C value is the inverse of the regularization strength, used to set how high the weights should be for the training data. Regularization shrinks the calculated coefficients of the least important variables to zero to reduce the number of variables, improving accuracy. The text model for Logistic Regression was also tuned with the above parameters as well as an additional parameter- the maximum number of iterations. This is the number of iterations the model is required to make in order to converge.

6.3.2 Random Forest Classifier

The parameters that were tuned for the random forest speech classifier were the maximum depth, maximum features, minimum samples to split, the number of estimators and whether or not to use bootstrapping. The maximum depth is the depth limit for each Decision Tree to be constructed and the maximum features is the number of features to consider for splitting according to the GINI impurity or entropy between variables. Similarly, the minimum samples for splitting is the lowest number of samples that must be used to split the variables and form the Decision Tree. Lastly, the number of estimators is the number of Decision Trees to be constructed, and bootstrapping determines how the samples are selected for the entire process. This parameter was set to false for the speech model in order to use the entire training dataset to construct the Decision Trees. The same parameters

were tuned for the text model, and both models used the GINI impurity value as the criteria to select variables.

6.3.3 Support Vector Machine

The parameters involved for tuning the text based Support Vector Classifier were the gamma and c values, the type of kernel to be used and whether or not to use probability estimates. The kernel type is a function used to lower complexity when constructing the decision boundary. The gamma value is the kernel coefficient, which determines the influence that a single training example has on the performance while c is the regularization parameter as mentioned for earlier models. The higher the value of gamma, the closer the points must be to affect one another. The speech based SVC was tuned on extra parameters such as the shape off the decision function, the degree of the kernel function if polynomial kernels are used and the shrinking parameter. Which is a heuristic that shortens training time.

6.3.4 Multinomial Naive Bayes

Only one parameter was tuned for both the speech and text Multinomial Naive Bayes model. This parameter was the alpha value, which is the degree of smoothing. Smoothing involves Laplace transformations that resolve the problem of zero probabilities, a phenomenon where the calculated probabilities will equal zero because there are no occurrences of a certain feature. Higher values of alpha assign increasingly uniform probabilities for each variable.

6.3.5 XGBoost

The speech based XGBoost model was tuned on parameters such as the gamma value which determines the splitting to construct the decision trees, the learning rate, the maximum depth for each decision tree, the number of estimators, lambda, which is the type of regularization to be used, and subsampling ratio, is the amount of data to be randomly sampled between iterations to prevent overfitting. The text based model was tuned on an additional parameter, which is the ratio of column wise subsampling for each decision tree.

6.3.6 MultiLayer Perceptron

The parameters which were tuned for the speech based MLP classifier were the learning rate, alpha value and maximum iterations. The learning rate is the amount by which weights are updated. Small values of the learning rate update the weights by small amounts, increasing training time while large values cause drastic changes in weights leading to an unstable neural network. The alpha value is the regularization strength to remove irrelevant variables and the maximum iterations are the number of epochs to use to iterate through until convergence. The text based MLP classifier was tuned on the same parameters with additional ones such as the number of hidden layers and the Adam optimizer to reduce loss and improve accuracy.

Chapter 7

Experimental Results & Discussion

The performance of each model after carrying out a 5-fold cross-validation is given in Table 7.1. We find that each ensemble method outperforms the base models, with the stacking ensemble (E4) giving the highest score of 81.2%. As the dataset is imbalanced, weighted accuracy (WA) was chosen as the core metric to evaluate the models. Apart from that, the macro-averaged F1 scores for each model is used as a secondary metric.

Model	**WA(%)**		**F1(%)**	
	Text	**Speech**	**Text**	**Speech**
Logistic Regression	65.2	63.3	66.2	63.2
MLP	65.2	64.5	65.9	64.0
Naive Bayes	63.8	46.9	64.6	46.4
Random Forest	63.5	61.5	64.7	68.6
SVM	62.6	67.3	64.3	67.0
XGBoost	63.6	68.4	64.7	68.6
E1 (Hard Voting)	74.9		75.5	
E2 (Soft Voting)	78.7		79.5	
E3 (Blending)	78.9		79.0	
E4 (Stacking)	**81.2**		**81.5**	

Table 7.1: Performance of each model (WA and F1 scores)

As seen in Table 7.1, both blending (E3) and stacking (E4) outperform the voting methods judging by weighted accuracy, which shows that the use of a meta model in combining classifiers can yield better results in an ensemble system. Stacking gives a better result compared to blending which can be explained by its use of the k-fold approach, allowing the meta classifier to be trained on a larger portion of the dataset as opposed to a hold-out validation set only. The confusion matrix for E4 has been given in Figure 7.1.

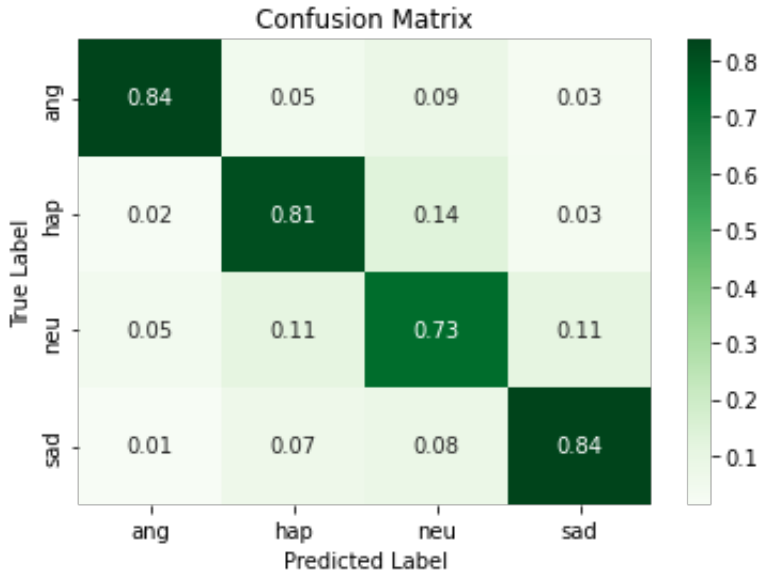

Figure 7.1: Normalized Confusion Matrix showing per-class accuracy of E4

The confusion matrix shows similar accuracies for the sad, happy, and angry classes. The neutral class tends to be misclassified as happy or sad, giving it a lower score. This could be due to the fact that while other emotions may have more pronounced features and characteristics, the neutral class does not. As this class signifies the absence of any specific emotion, the classification task may be more difficult due to its ambiguous nature.

Reference	**WA(%)**	**UA(%)**
Tripathi (2019)[8]	69.5	76.1
Cai (2019)[12]	70.4	71.3
Xu (2020)[16]	70.4	69.5
Mustaqeem (2020)[22]	-	72.3
Makiuchi (2021)[23]	73.5	73
Zheng (2019)[18]	75.0	75.0
Atmaja (2019)[10]	-	75.5
Yoon (2019)[17]	76.5	77.6
Lian (2020)[21]	82.7	-
Proposed (E4)	**81.2**	**80.8**

Table 7.2: Comparison with state-of-the-art results for classifying four emotions on the IEMOCAP dataset

In Table 7.2, we see our stacking ensemble (E4) outperforms most implementations from recent years tested on four emotions on IEMOCAP. Papers using a multimodal approach such as [15], [12], [23], and [21] focus on combining individual models for text and speech while incorporating deep learning methods. Our results show that using a larger number of models for each modality leads to better performance. Furthermore, we confirm that simpler machine learning models in an ensemble system can perform on par with state-of-the-art methods.

Chapter 8

Limitations & Improvements

Despite the impressive results achieved, we faced a number of obstacles during our experiments. As the dataset we used was highly imbalanced, all classes of emotions could not be used. After dropping ambiguous categories, merging labels for similar emotions, and performing other preprocessing steps, we were left with only 5531 samples. The training dataset we worked with was not large enough for this reason. This could be improved upon if we implemented some manner of data augmentation or upsampling to increase the size of our dataset. Another drawback is that in some cases, transcriptions alone did not seem to match the emotion labeled beside it, making the associated label given by the annotators inaccurate during training of text models. Moreover, the proposed implementation assumes that speech-to-text conversion is done with a high accuracy to produce text data. However, this may not always be possible for real life usage. Training the models on a larger text corpus could improve them, especially for generalized usage.

Chapter 9

Conclusion

In this paper, we propose a model combining commonly used classifiers to tackle the task of emotion recognition. A multimodal approach is taken where six base classifiers are trained on speech and text data separately and put together in a single model. We also investigate the performance of four different ensemble techniques. Our findings show that each ensemble performs better than the individual base models, with the stacking ensemble giving the highest accuracy of 81.2% which surpasses previous research on the same dataset. For future work, deep learning classifiers can be incorporated in the ensemble and more datasets can be tested. Additionally, visual information as a third modality can be used to enhance the proposed model.

Bibliography

[1] D. H. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, no. 2, pp. 241–259, 1992, ISSN: 0893-6080. DOI: https://doi.org/10.1016/S0893-6080(05)80023-1. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608005800231.

[2] C. Busso, M. Bulut, C.-C. Lee, *et al.*, "Iemocap: Interactive emotional dyadic motion capture database," *Journal of Language Resources and Evaluation*, vol. 42, no. 5, pp. 335–359, Dec. 2008.

[3] J. Sill, G. Takács, L. W. Mackey, and D. Lin, "Feature-weighted linear stacking," *ArXiv*, vol. abs/0911.0460, 2009.

[4] P.-Y. Shih, C.-P. Chen, and C.-H. Wu, "Speech emotion recognition with ensemble learning methods," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 2756–2760. DOI: 10.1109/ICASSP.2017.7952658.

[5] S. R. Livingstone and F. A. Russo, "The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english," *PLOS ONE*, vol. 13, no. 5, pp. 1–35, May 2018. DOI: 10.1371/journal.pone.0196391. [Online]. Available: https://doi.org/10.1371/journal.pone.0196391.

[6] K. Noh, J. Lim, S. Chung, G. Kim, and H. Jeong, "Ensemble classifier based on decision-fusion of multiple models for speech emotion recognition," Oct. 2018, pp. 1246–1248. DOI: 10.1109/ICTC.2018.8539502.

[7] M.-H. Su, C.-H. Wu, K.-Y. Huang, and Q.-B. Hong, "Lstm-based text emotion recognition using semantic and emotional word vectors," *2018 First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia)*, pp. 1–6, 2018.

[8] S. Tripathi and H. S. M. Beigi, "Multi-modal emotion recognition on iemocap dataset using deep learning," *ArXiv*, vol. abs/1804.05788, 2018.

[9] 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.

[10] B. T. Atmaja, K. Shirai, and M. Akagi, "Speech emotion recognition using speech feature and word embedding," in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2019, pp. 519–523. DOI: 10.1109/APSIPAASC47483.2019.9023098.

[11] E. Batbaatar, M. Li, and K. H. Ryu, "Semantic-emotion neural network for emotion recognition from text," *IEEE Access*, vol. 7, pp. 111 866–111 878, 2019. DOI: 10.1109/ACCESS.2019.2934529.

[12] L. Cai, Y. Hu, J. Dong, and S. Zhou, "Audio-textual emotion recognition based on improved neural networks," *Mathematical Problems in Engineering*, vol. 2019, pp. 1–9, Dec. 2019. DOI: 10.1155/2019/2593036.

[13] D. Haryadi and G. Putra, "Emotion detection in text using nested long short-term memory," *International Journal of Advanced Computer Science and Applications*, vol. 10, Jan. 2019. DOI: 10.14569/IJACSA.2019.0100645.

[14] G. Sahu, "Multimodal speech emotion recognition and ambiguity resolution," *CoRR*, vol. abs/1904.06022, 2019. arXiv: 1904.06022. [Online]. Available: http://arxiv.org/abs/1904.06022.

[15] S. Tripathi, A. Kumar, A. Ramesh, C. Singh, and P. Yenigalla, "Deep learning based emotion recognition system using speech features and transcriptions," *ArXiv*, vol. abs/1906.05681, 2019.

[16] H. Xu, H. Zhang, K. Han, Y. Wang, Y. Peng, and X. Li, "Learning alignment for multimodal emotion recognition from speech," in *INTERSPEECH*, 2019.

[17] S. Yoon, S. Byun, S. Dey, and K. Jung, "Speech emotion recognition using multi-hop attention mechanism," *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2822–2826, 2019.

[18] C. Zheng, C. Wang, and J. Ning, "An ensemble model for multi-level speech emotion recognition," *Applied Sciences*, vol. 10, p. 205, Dec. 2019. DOI: 10.3390/app10010205.

[19] N. T. Ira and M. O. Rahman, "An efficient speech emotion recognition using ensemble method of supervised classifiers," in *2020 Emerging Technology in Computing, Communication and Electronics (ETCCE)*, 2020, pp. 1–5. DOI: 10.1109/ETCCE51779.2020.9350913.

[20] J. Joy, A. Kannan, S. Ram, and S. Rama, "Speech emotion recognition using neural network and mlp classifier," 2020.

[21] Z. Lian, J. Tao, B. Liu, J. Huang, Z. Yang, and R. Li, "Context-dependent domain adversarial neural network for multimodal emotion recognition," in *INTERSPEECH*, 2020.

[22] Mustaqeem, M. Sajjad, and S. Kwon, "Clustering-based speech emotion recognition by incorporating learned features and deep bilstm," *IEEE Access*, vol. 8, pp. 79 861–79 875, 2020. DOI: 10.1109/ACCESS.2020.2990405.

[23] M. R. Makiuchi, K. Uto, and K. Shinoda, "Multimodal emotion recognition with high-level speech and text features," *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 350–357, 2021.

[24] W. Zehra, A. R. Javed, Z. Jalil, H. U. Khan, and T. R. Gadekallu, "Cross corpus multi-lingual speech emotion recognition using ensemble learning," *Complex & Intelligent Systems*, pp. 1–10, 2021.