

# USING CONVOLUTIONAL NEURAL NETWORKS FOR NATURALISTIC SPEECH EMOTION RECOGNITION (N-SER)

A Final Defense  
Presented to  
the Faculty of the College of Computer Studies  
De La Salle University Manila

In Partial Fulfillment  
of the Requirements for the Degree of  
Bachelor of Science in Computer Science

by

CHU, Chuan-chen  
DELIMA Jr., Reynaldo K.  
JATICO II, Nilo Cantil K.  
TAN, Jedwig Siegfried S.

Merlin Teodosia Suarez  
Adviser

July 7, 2022

## Abstract

Speech emotion recognition is prevalent in the field of human-computer interactions. It helps machines understand emotions that are applicable to chat-bots and virtual personal assistants. Findings from literature indicate that data-driven approaches such as Convolutional Neural Networks (CNN) were used to create models using data from posed emotion expressions. Convolutional Neural Networks are used in studies such as image processing and speech emotion recognition. However, the use of posed and small-scale datasets result in models overfitting. In this study, Mel-frequency cepstral coefficients (MFCC) and Mel-Spectrogram features were extracted from the naturalistic data of the IEMOCAP dataset. These audio features are used as input for the experiments. Traditional machine learning models such as Random Forest Classifier, XGBoost Classifier, Multi-Layer Perceptron Classifier, Support Vector Machine and Logistic Regression were used for the experiments. Mel-spectrogram features as input has the highest performance among the machine learning models with a validation accuracy of 77.7%. Furthermore, deep learning models such as TDNN, CNN with LSTM, Base CNN and ResNet50 were also used for the experiments. The TDNN model using Convolution layers with MFCC as input reached a validation accuracy of 70.2%. All models were validated using cross-validation with performance metrics. Results show that machine learning models perform better with Mel-spectrogram as input.

**Keywords:** Affective Computing, Speech emotion recognition, Machine learning, Deep Learning, Naturalistic datasets, End-to-end model

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Background of the Study . . . . .	1
1.2	Research Objectives . . . . .	6
1.2.1	General Objective . . . . .	6
1.2.2	Specific Objectives . . . . .	6
1.3	Scope and Limitations of the Study . . . . .	6
1.4	Significance of the Study . . . . .	7
1.5	Methodology . . . . .	8
1.5.1	Model Architecture . . . . .	8
1.5.2	Model Validation . . . . .	9
<b>2</b>	<b>Review of Related Literature</b>	<b>10</b>
2.1	Emotion Datasets . . . . .	10
2.2	Posed Datasets . . . . .	11
2.3	Induced Datasets . . . . .	12
2.4	Naturalistic Datasets . . . . .	13
2.5	Datasets and Emotion Models . . . . .	15
2.6	Machine Learning in Speech Emotion Recognition . . . . .	17

<b>3</b>	<b>Theoretical Framework</b>	<b>22</b>
3.1	Emotion Theory . . . . .	22
3.1.1	Categorical Approach . . . . .	22
3.1.2	Dimensional Approach . . . . .	25
3.1.3	Appraisal Approach . . . . .	28
3.2	Signal Processing . . . . .	28
3.2.1	Speech in Emotion Recognition . . . . .	29
3.2.2	Fourier Transform . . . . .	29
3.2.3	Discrete Cosine Transform . . . . .	31
3.2.4	MFCC . . . . .	32
3.2.5	Spectrogram . . . . .	33
3.3	End-to-End Model Framework . . . . .	34
3.4	Convolutional Neural Networks . . . . .	34
3.4.1	Hidden Layer . . . . .	35
3.4.2	Output Layer . . . . .	38
3.5	Time Delay Neural Network . . . . .	39
<b>4</b>	<b>Building a Speech Emotion Recognition Model</b>	<b>41</b>
4.1	Framework . . . . .	41
4.1.1	Input . . . . .	42
4.1.2	Training and Testing Sets . . . . .	42
4.1.3	Classification . . . . .	42
4.1.4	Output . . . . .	43
4.2	Dataset Details . . . . .	45

4.2.1	Dataset Contents and Environment . . . . .	45
4.2.2	Emotion Labels and Distribution . . . . .	46
4.2.3	Pre-processing and Feature Extraction . . . . .	47
4.2.4	Dataset Features . . . . .	49
4.2.5	Libraries and Tools . . . . .	50
4.3	Traditional Machine Learning . . . . .	52
4.3.1	Random Forest Classifier . . . . .	53
4.3.2	XGBoost Classifier . . . . .	53
4.3.3	Multi-Layer Perceptron Classifier . . . . .	54
4.3.4	Support Vector Machine . . . . .	55
4.3.5	Logistic Regression . . . . .	55
4.4	Model Architecture . . . . .	56
4.4.1	ResNet50 . . . . .	56
4.4.2	TDNN . . . . .	57
4.4.3	CNN + LSTM . . . . .	59
4.4.4	Base CNN . . . . .	60
<b>5</b>	<b>Results and Analysis</b>	<b>62</b>
5.1	Random Forest Classifier . . . . .	64
5.2	XGBoost Classifier . . . . .	65
5.3	Multi-Layer Perceptron Classifier . . . . .	66
5.4	Support Vector Machine . . . . .	67
5.5	Logistic Regression . . . . .	68
5.6	Analysis of Machine Learning Models . . . . .	68

5.7	ResNet50 NSER . . . . .	71
5.8	TDNN NSER . . . . .	72
5.9	CNN + LSTM . . . . .	77
5.10	Base CNN . . . . .	79
5.11	Further Analysis of Deep Learning Models . . . . .	81
5.12	Discussion . . . . .	84
<b>6</b>	<b>Conclusion</b>	<b>89</b>
6.1	Recommendations . . . . .	90
	<b>References</b>	<b>92</b>
<b>A</b>	<b>Research Ethics Documents</b>	<b>102</b>
<b>B</b>	<b>Similarity Report</b>	<b>114</b>
<b>C</b>	<b>Summary of Models For Literature</b>	<b>116</b>
<b>D</b>	<b>Full Experiment Results of Machine Learning Models</b>	<b>129</b>
D.1	Random Forest Classifier Parameter Experiments . . . . .	130
D.2	XGBoost Classifier Parameter Experiments . . . . .	135
D.3	Multi Layer Perception Classifier . . . . .	140
D.4	Support Vector Machines . . . . .	143
D.5	Logistic Regression . . . . .	144
<b>E</b>	<b>Comparison of Deep Learning Models in Present Literature With This Study</b>	<b>146</b>
E.1	TDNN Architecture . . . . .	147

E.2	ResNet50 Architecture . . . . .	150
E.3	CNN+LSTM Architecture . . . . .	152
E.4	CNN Architecture . . . . .	154
<b>F</b>	<b>Resource Persons</b>	<b>157</b>

# List of Figures

3.1	Visualization of Ekman's Theory (Ekman, 2021) . . . . .	23
3.2	Visualization of the Six Basic Emotions (Ekman, 2021) . . . . .	23
3.3	Table of Action Units for Upper and Lower facial expressions (De la Torre et al., 2015) . . . . .	24
3.4	Table of AU subsets for each basic emotion (Ghayoumi, Bansal, 2016)	25
3.5	Circumplex model (Ekman, 2021) . . . . .	26
3.6	Positive activation negative activation model (Reyes et al, 2019) . . .	27
3.7	Wheel of emotions model (Six Seconds, 2021) . . . . .	27
3.8	Visualization of a frequency-domain representation of a signal (Chaudhary, 2020) . . . . .	30
3.9	Cepstrum Conversion Formula (Elelu, 2021) . . . . .	32
3.10	Visualization of a spectrogram from a given time frame (Chaudhary, 2020)	33
3.11	Visualization of three sample mel spectrogram (Sabra, 2021) . . . . .	34
3.12	Convolutional Neural Network Architecture (Saha, 2018) . . . . .	35
3.13	Computation of TDNN with sub-sampling (red) and without sub-sampling (blue + red) (Peddinti, Povey & Khudanpur, 2015) . . . . .	39
4.1	Machine Learning Framework for Speech Emotion Recognition . . . . .	42
4.2	Histogram for Number of Utterances per length . . . . .	45



4.3	Naturalistic Emotion Label Distribution Busso et. al., 2008 . . . .	47
4.4	An Example of a padded audio clip . . . . .	47
4.5	An Example of a padded audio clip for 10 seconds . . . . .	48
4.6	An Example of a clipped audio clip . . . . .	48
4.7	An Example of a padded audio clip . . . . .	50
4.8	Mel-Spectrogram Feature Extraction Pipeline . . . . .	51
4.9	MFCC Feature Extraction Pipeline . . . . .	52
4.10	Resnet-50 Architecture (Manaswi, 2018) . . . . .	56
4.11	TDNN Dense Model Architecture . . . . .	57
4.12	TDNN Convolution Model Architecture . . . . .	58
4.13	CNN + LSTM Model Architecture . . . . .	59
4.14	Base CNN Model Architecture for MFCC . . . . .	60
4.15	Base CNN Model Architecture for Mel-Spectrogram . . . . .	61
5.1	Visualization of Sample Ideal Graph of Accuracy and Loss . . . .	64
5.2	Random Forest Classifier Confusion Matrix . . . . .	64
5.3	XGBoost Classifier Confusion Matrix . . . . .	65
5.4	MLP Confusion Matrices . . . . .	66
5.5	SVM Confusion Matrices . . . . .	67
5.6	LR Confusion Matrices . . . . .	68
5.7	ResNet50 NSER MFCC Performance Results . . . . .	71
5.8	ResNet50 NSER MFCC Performance Results . . . . .	71
5.9	TDNN Dense NSER MFCC Performance Results . . . . .	72
5.10	TDNN Dense NSER Mel-Spectrogram Performance Results . . . .	73

5.11	TDNN Convolution NSER MFCC Performance Results . . . . .	73
5.12	TDNN Convolution NSER Mel-Spectrogram Performance Results	74
5.13	TDNN Dense NSER Confusion Matrices . . . . .	74
5.14	TDNN Convolution NSER Confusion Matrices . . . . .	75
5.15	CNN+LSTM NSER MFCC Performance Results . . . . .	77
5.16	CNN+LSTM NSER Mel-Spectrogram Performance Results . . . .	77
5.17	CNN+LSTM NSER Confusion Matrices . . . . .	78
5.18	Base CNN NSER MFCC Performance Results . . . . .	79
5.19	Base CNN NSER Mel-spectrogram Performance Results . . . . .	79
5.20	Base CNN NSER Confusion Matrices . . . . .	80
5.21	Happy Comparison with Neutral Utterance . . . . .	85
5.22	Sad Comparison with Neutral Utterance . . . . .	86

# List of Tables

2.1	Summary of Posed Datasets . . . . .	12
2.2	Summary of Induced Datasets . . . . .	13
2.3	Summary of Naturalistic Datasets . . . . .	15
5.1	Machine Learning Experiments 10s Vs 29s MFCC . . . . .	69
5.2	Machine Learning Experiments 10s Vs 29s Mel Spectrogram . . .	70
5.3	Deep Learning Experiments 10s Vs 29s MFCC . . . . .	81
5.4	Deep Learning Experiments 10s Vs 29s Mel-Spectrogram . . . . .	82
5.5	Number of Train and Test Data for Clipped and Unclipped . . . .	83
5.6	Ratio of Recognized Clipped and Unclipped Test Data using MFCC as Input . . . . .	83
5.7	Ratio of Recognized Clipped and Unclipped Test Data using Mel- Spectrogram as Input . . . . .	84
5.8	Best DL/ML model for MFCC 10s . . . . .	87
5.9	Best DL/ML model for MFCC 29s . . . . .	87
5.10	Best DL/ML model for Mel 10s . . . . .	87
5.11	Best DL/ML model for Mel 29s . . . . .	87
C.1	Summary of Models. . . . .	117

D.1	Results With 10 Audio Sec. Clips Using MFCC Features . . . . .	130
D.2	Results With 30 Audio Sec. Clips Using MFCC Features . . . . .	131
D.3	Results With 10 Audio Sec. Clips Using Mel-spectrogram Features	133
D.4	Results With 30 Audio Sec. Clips Using Mel-spectrogram Features	134
D.5	Results Using MFCC Features . . . . .	136
D.6	Results Using Mel-Spectrogram Features . . . . .	138
D.7	Multi Layer Perception Classifier Results on Different Activation Using MFCC Features . . . . .	140
D.8	Multi Layer Perception Classifier Results on Different Solver Using MFCC Features . . . . .	141
D.9	Multi Layer Perception Classifier Results on Different Learning Rate Schedules Using MFCC Features . . . . .	141
D.10	Multi Layer Perception Classifier Results on Different Initial Learn- ing Rates Using MFCC Features . . . . .	141
D.11	Multi Layer Perception Classifier Results on Different Activation Using Mel-spectrogram Features . . . . .	142
D.12	Multi Layer Perception Classifier Results on Different Solver Using Mel-spectrogram Features . . . . .	142
D.13	Multi Layer Perception Classifier Results on Different Learning Rate Schedules Using Mel-spectrogram Features . . . . .	143
D.14	Multi Layer Perception Classifier Results on Different Initial Learn- ing Rates Using Mel-spectrogram Features . . . . .	143
D.15	Support Vector Machine Performance Results on Different Kernel Types using MFCC Features . . . . .	144
D.16	Support Vector Machine Performance Results on Different Kernel Types using Mel-spectrogram Features . . . . .	144
D.17	Logistic Regression Performance Results on Different Solvers using MFCC Features . . . . .	144

D.18	Logistic Regression Performance Results on Different Solvers using Mel-spectrogram Features . . . . .	145
E.1	Performance of Models in Present Studies Using TDNN . . . . .	147
E.2	Performance of Models in Present Studies Using ResNet50 . . . . .	150
E.3	Performance of Models in Present Studies Using CNN+LSTM . . . . .	153
E.4	Performance of Models in Present Studies Using CNN . . . . .	154

# Chapter 1

## Introduction

This section introduces the state of the art in the field of speech emotion recognition. This is followed by a presentation of the objectives, scope and significance of the study. The chapter concludes with the contribution of the study in the field of speech emotion recognition.

### 1.1 Background of the Study

Affective computing is the research and application of systems and machines that recognize and determine emotions, or express it themselves. With emotions being an integral part of human understanding and comprehension, the application of affective computing in daily life can help make machines much more connected to their users, as well as enable them to build and enact more humanized decisions (Picard, 2000). The attempt to bridge and naturalize interactions between human-computer interactions in the field of affective computing has led to physiologists and scientists alike to read and conduct studies on emotions by analyzing elements of emotions from 3 distinct areas, namely in facial expressions, voice or physiological signals from a certain individual (Sebe et al., 2005).

The study of emotions through facial expressions has been studied since the 1970s, with the findings of Paul Ekman and his research team noting that emotions commonly associated with ubiquitous facial expressions and gestures include happiness, sadness, anger, fear, surprise, and disgust (Ekman, 1994). A psychologist by the name of David Matsumoto would then add another emotion commonly exhibited in facial expressions, namely contempt (Matsumoto, Takeuchi, Andayani, Kouznetsova, & Krupp, 1998). The taxonomy of facial expressions was first in-

troduced by Paul Ekman and Wallace Friesen who invented the Facial Action Coding System which coded facial expressions based on facial movements into Action Units that are based on muscular features. This is done manually through still images (Ekman & Friesen, 1978). The general overview of emotion recognition via facial recognition involves either methods used via images or videos as input data. Facial image process involves the extraction of features from images that are then given to a classification system that gives a predefined emotion as an output. In video image processing the extraction of features is divided into mainly two methods, one that attempts to find and follow specific details and features in the video input data, while the other application is through the use of region-based applications that focus on specific parts of the face, including the lips (Sebe et al., 2005).

Social interactions rely also on nonverbal emotional displays to communicate social emotions and inform people about their feelings (Reed et al., 2020). There are a growing number of studies that the body plays a role in emotional communication via body posture and movement. With the continual development of affective computing, learning emotions through body movement is possible with studies using motion capture systems (M. Zhang et al., 2020) to capture information, which will be utilized for machine learning.

Speech has been the primary medium of human interaction. Speech itself conveys information that contains emotion cues as a form of communication between humans (Ntalampiras, 2021). Dialogues are conducted through speech, and with it comes the universal ability of humans to recognize emotions through voice. With the development of technology, the field of affective computing emerged as it addresses the utilization of machines in recognizing human emotion through different modalities such as speech (B. W. Schuller, 2018). Prosodic or vocal features may be used to determine the emotion being expressed through audio by utilizing speech analysis methods (Breazeal & Aryananda, 2002).

The first study on the field of speech emotion recognition emerged during a 1996 research conducted by students of Carnegie Mellon University (Dellaert, Polzin, & Waibel, 1996). At that time, the concept of speech emotion recognition was defined as the machine’s ability to automatically recognize human emotions and affective states through speech (B. W. Schuller, 2018). The field of speech emotion recognition is a developing field that could be the future of human to computer interactions and help technology achieve a higher level of understanding of human depth and complexity. Speech-based chatbot technology has been presented as a way to provide conversational agents to the users (du Preez, Lall, & Sinha, 2009). Together with this technology, virtual personal assistants (VPA) such as Siri, Alexa and Cortana have been steadily rising in popularity. VPA has the ability to assist people in their everyday lives with commands through voice

input such as weather and news updates. However, it is not proficient enough to accurately recognize emotions (Venkataramanan & Rajamohan, 2019).

The task of achieving a natural interaction between humans and machines through recognizing human emotion is complicated, as different individuals express emotions in different ways (Tzirakis, Nguyen, Zafeiriou, & Schuller, 2021). In light of this, studies and researches in the field of affective computing aim to close the gaps that exist in human computer interaction for the purpose of producing effective methods of automatic emotion recognition (B. W. Schuller, 2018). Such studies that apply affective computing in human interaction include its application in online or distance learning, such as a proposal by X.W. Wang, and Z.L. Wang for a Client/Server E-learning system that uses artificial psychology in order for teachers to monitor the emotional states of their students (J. Liu, Tong, Han, Yang, & Chen, 2013). A study from Yale University compared voice-only modality against visual-only and audiovisual modality in terms of empathic accuracy in recognizing emotion. The study showed that a voice-only modality performs on a higher level in regards to humans' empathic accuracy as compared to visual-only or audiovisual communication (Kraus, 2017).

Multiple speech emotion recognition models have been conceived and developed in the field of technology (Sun, 2020). Speech emotion recognition models utilize para-linguistic or semantic features extracted from speech in order to perform speech emotion recognition tasks (Tzirakis et al., 2021). Speech features play a major factor in determining the performance of the model (Venkataramanan & Rajamohan, 2019; Ntalampiras, 2021). As such, feature extraction is a major part in a speech emotion recognition process because the quality of feature extraction affects the accuracy of speech emotion recognition (Huang, Gong, Fu, & Feng, 2014).

Studies on speech emotion recognition utilizes speech data gathered from controlled or naturalistic environments. There are three defined categories of speech emotion recognition datasets, with each dataset containing either audio or audiovisual clips. *Posed* datasets are attributed to being entirely staged, with actors usually stating certain sentences in the emotional context given to them. *Naturalistic* datasets, on the other hand, contain data captured through genuine interaction among the participants as they display authentic emotion. The data gathered in naturalistic datasets are not acted, and oftentimes are collected recordings of television shows (Wu, Falk, & Chan, 2011; Grimm, Kroschel, & Narayanan, 2008). An instance of a naturalistic dataset is audiovisual data collected from a set of participants who watched a commercial and discussed it with their partner afterwards (Tzirakis et al., 2021; Busso et al., 2008). Finally, *induced* datasets are not purely staged, and are in line with being produced in a controlled environment, usually involving sessions with other speakers for a more natural characteristic



(Abbaschian, Sierra-Sosa, & Elmaghraby, 2021).

The emotions gathered in the speech emotion recognition datasets may be labeled based on different emotion theories: *categorical* or *dimensional* labels. Categorical labels imply that emotions can be characterized into different basic or fundamental emotions (Jeon, 2017). In this context, the six basic types of emotions are used: fear, anger, happiness, disgust, sadness, and surprise (Jeon, 2017). An example of datasets containing audio data that uses discrete emotion theory to label the data is CAISA (Sun, 2020). Other datasets with discrete labels are the Berlin Emotional Speech Database (Burkhardt, Paeschke, Rolfes, Sendlmeier, & Weiss, 2005), Ryerson Audio-Visual Database of Emotional Speech and Song, Toronto Emotional Speech Set (Venkataramanan & Rajamohan, 2019), Friedrich-Alexander-Universität Aibo Emotion Corpus (Cao, Verma, & Nenkova, 2015), and Interactive Emotional Dyadic Motion Capture (Abbaschian et al., 2021).

Dimensional labels are based on the dimensional emotion theory, this theory states that emotions can be presented as points within a continuous space (Jeon, 2017). Valence and Arousal are usually used as dimensions in a two-dimensional model, while an additional dimension such as Dominance can be added to form a three-dimensional space that can define an emotion (Jeon, 2017). An emotion dataset that utilized dimensional emotion theory for labeling emotion is the Sentiment Analysis in the Wild (SEWA) database (Tzirakis et al., 2021). This particular dataset is naturalistic and collected from a set of participants that were tasked to watch a commercial and then discuss it with their partners afterwards (Tzirakis et al., 2021). Another dataset that utilized dimensional labels is the ‘Vera Am Mittag’ (VAM) dataset, which contains audiovisual recordings of a German talk show of the same name (Wu et al., 2011)).

Features are extracted from speech data and used to create speech emotion recognition models. Common features that are extracted are prosodic features such as pitch, as well as energy, duration, formant, Mel frequency cepstrum coefficient, and linear prediction cepstral coefficient (Ingale & Chaudhari, 2012). However, selecting proper speech features to represent corresponding emotion categories is an issue that exists in speech emotion recognition (Sun, 2020). Due to this issue, it remains unclear which features are more effective in representing speech emotion (Mirsamadi, Barsoum, & Zhang, 2017). Additionally, most acoustic features from feature extraction may not be discriminative enough to identify subjective emotions (Sun, 2020; Sezgin, Gunsel, & Kurt, 2012). Consequently, utilization of raw speech signal without the process of acoustic feature selection is implemented in some studies to prevent artificial intervention from omitting emotion information that cannot be mathematically modelled (Sun, 2020; Trigeorgis et al., 2016).

Speech emotion recognition models utilize different methods. Deep Neural Networks (DNN) are used in studies for the purpose of modelling emotions using multiple modalities (Tzirakis, Trigeorgis, Nicolaou, Schuller, & Zafeiriou, 2017). However, various limitations are present in the works and contributions in the field of speech emotion recognition.

Particular limitations exist in the process of data labeling as it is expensive and time-consuming in large quantities as the process requires an expert’s knowledge (Deng, Xu, Zhang, Frühholz, & Schuller, 2018). Another limitation exists in feature extraction, as it is unclear which speech features best represent emotions (Mirsamadi et al., 2017). A study that used a Residual Convolutional Neural Network for speech emotion recognition used raw speech data to address the issue of data labeling and feature extraction (Sun, 2020). The utilization of R-CNN on raw speech data was evaluated on three databases which are CAISA, EMODB, and IEMOCAP, and the results showed that the algorithm showed higher accuracy results compared to other speech emotion recognition algorithms (Sun, 2020). However, the algorithm was unable to work in real-time, and it was computationally expensive so it could not be integrated on a mobile device (Sun, 2020).

Studies have also been conducted in order to improve current methods and frameworks. As an example, the usage of dynamic weights has generally resulted in increased accuracy rates and a lower transmission loss than with static weights (Chae, Kim, Shin, Kim, & Lee, 2018). The usage of Convolutional Neural Networks has been present in more recent studies, and with the help of various breakthroughs in the field of speech emotion recognition, may in the near future do away with the need for manual processing of features, as well as be able to improve upon using additional modalities, and improve the nature and efficacy of speech emotion recognition as a field.

End-to-end models are used in automatic speech recognition studies as they provide a simplified structure for speech recognition process compared to a traditional automatic speech recognition pipeline (Wang & Li, 2019). Similarly, end-to-end models are recently utilized in speech emotion recognition studies (Chae et al., 2018). It is defined as a type of model that simplifies a traditional speech emotion recognition pipeline as it directly maps the input signal into corresponding mapped output (Sun, 2020). As such, there is no feature extraction in some speech emotion recognition models (Chae et al., 2018). An advantage of an end-to-end model is its simplified structure compared to conventional pipeline of a traditional speech emotion recognition system where feature extraction is a separate method from decoding features through an acoustic model (Sun, 2020). In contrast, each process in the conventional pipeline is independently constructed, and therefore independently optimized (Sainath, 2020).

Speech emotion recognition studies that used posed datasets have relatively higher accuracy results compared to studies that used naturalistic datasets. Posed datasets such as CAISA has an average accuracy of 81.67%, EMODB with 76.16%, RAVDES with 68.99% and TESS with 57.86%. In comparison, the highest speech emotion recognition accuracy result yielded from a naturalistic dataset was 64.86% (Abbaschian et al., 2021). Another study comparing speech emotion recognition in acted and spontaneous context showed that a Hidden Markov Model classifier has higher emotion recognition accuracy in for acted context compared to spontaneous or naturalistic context (Chenchah & Lachiri, 2014).

## **1.2 Research Objectives**

### **1.2.1 General Objective**

The study aims to develop a speech emotion recognition model to recognize naturalistic emotions.

### **1.2.2 Specific Objectives**

1. Review the literature on the nature of emotion datasets.
2. Review the literature on speech emotion recognition models.
3. Design a speech emotion recognition model that recognizes emotions in a realistic context.
4. Evaluate the performance of the model using established metrics.

## **1.3 Scope and Limitations of the Study**

The emotional datasets will be reviewed specifically on the nature of the dataset. Posed, induced and naturalistic datasets are the three types of emotional datasets. The datasets will be reviewed based on its origin, speaker count, the size of the audio files, the description of the dataset, and the emotional labels will be investigated.

The study reviews the various processes of speech emotion recognition models such as data pre-processing, feature extraction, training, and testing of data. The metric used in determining the performance of the model based on accuracy will also be reviewed.

The speech emotion recognition model will be designed as an end-to-end model. The end-to-end model consists of methods such as data pre-processing through data cleaning and data reduction, training, and testing of data through the utilization of naturalistic datasets that are large-scale in nature.

The validation of the model will consist through technical which will be done through the usage of cross-validation and comparison within related works in the field of speech emotion recognition.

The criteria for the dataset in the study will be a dataset used in studies from 2016 or later, and the dataset is centered around the categorical or dimensional approach as the preferred emotion theory. Also, the dataset using naturalistic data, as to emulate emotions taken in a real-world context. IEMOCAP will be used as the preferred dataset for the study, using only the naturalistic portion of the dataset containing 4784 utterances split between 10 labels including *others* (Busso et al., 2008).

## 1.4 Significance of the Study

The study aims to design and evaluate an end-to-end speech emotion recognition model that is able to accurately detect emotions in a realistic context through the use of naturalistic datasets in model training and testing. The findings of this study will provide additional insights into the field of affective computing as it can contribute with the advancement of virtual personal assistants that can understand human emotions through voice interaction. Virtual personal assistants that can detect emotions through voice command and are able to react based on the recognized emotion can be developed with the use of the model in this study. Furthermore, the model will have relative versatility in terms of human-computer interaction because it can be utilized in multiple fields such as mental health and home assistance.

The model can be used to develop virtual agents that can potentially diagnose a human based on the expressed emotion. It can be utilized in various therapy or counseling clinics that can employ these models as virtual agents that can monitor the mood of a person, and help detect if they are in any emotional distress in order to avoid possible complications as a result of these emotions. Due to the model

being able to detect negative human emotions such as depression, it can contribute to people’s mental health by providing assistance through emotion recognition.

The possibility of utilizing this technology in home assistance systems, as well as in speech-based chat-bots to provide conversational agents to users can be explored. The prospect of having accurate emotion detection and proper feedback garnered a large amount of publicity and commendation from the public, most notably during the unveiling of a similarly developed product called the Olly in 2017, which was never able to be realized in a physical product due to apparent software deficiencies (Summers, 2020). The development of speech emotion recognition is vital as technology rapidly advances in society, as speech emotion recognition offers the capability to link technology to its user at a deeper level than what is currently possible, and is able to provide a more substantial and unique experience.

## **1.5 Methodology**

### **1.5.1 Model Architecture**

This section discusses the model architecture of each proposed model. A set of models was constructed and their performance was observed in terms of model accuracy. 3 different models was used based on the deep learning framework, with the study utilizing a 2D Residual Network Model, a Convolutional Neural Network with Long Short-Term Memory, and a Time-Delay Neural Network.

The proposed models incorporated both 40 MFCC feature set as well as Mel-Spectrogram feature set as separate inputs. Additionally, the data was also used into both 1D and 2D formats depending on the model used. The data that was used for the models was split into training and testing sets in a 80-20 ratio, with 80% of the data on training set and 20% on test set, and will be classified using emotions based on Ekman’s theory on categorical emotions.

Experiments was conducted using the proposed models and were compared with related works in the field of speech emotion recognition. This was done in order to identify which network architecture yields the best performance based on the accuracy. Separate experiments on each model was done using the same amount of data with the same set of emotion labels.

### 1.5.2 Model Validation

Model Validation is the process of acquiring validated data from models using different kinds of technique. Cross validation was the major method or technique used in validating the model in this research. Cross validation is a simple data re-sampling method to assess the generalization ability of the model and to also prevent over-fitting in the training process, models undergoing the method of cross validation has its dataset split into sets to ensure no over-fitting happening in the training process.

The model was validated using known performance metrics. The performance metrics consist of the F1 score, classification accuracy and logarithmic loss. The F1 score is a classifier that takes the harmonic mean of both the precision and recall metrics of the model, as is used as a singular metric that assesses model performance. Classification accuracy is an accuracy based on the ratio of the number of correct predictions to the total number of the input sample. The logarithmic loss indicates how close the recognition is to the corresponding value.

## Chapter 2

# Review of Related Literature

Multiple research articles and advancements on speech emotion recognition written from 2018 to 2021 are reviewed and analyzed to understand the current state of the field. In this chapter, the synthesis of literature is organized into two sections, emotion datasets and model comparison. Mainly, to discover the approaches tackled by different researchers contributes to understanding the different models and datasets used currently in the field.

There is a need to examine the various datasets available in the field since there are different types of datasets such as posed, induced and naturalistic. The comparison and the evaluation of the model is also important because the model determines the performance based on accuracy. The performance of emotion speech recognition models relies on features extracted from the audio (Venkataramanan & Rajamohan, 2019).

### 2.1 Emotion Datasets

This section contains datasets classifications based on the research article of (Abbaschian et al., 2021). Datasets may be classified based on the nature of their creation and how the emotional expressions were conveyed. In particular, datasets may be classified as posed, induced, or naturalistic (Abbaschian et al., 2021). The section will also contain a detailing of the datasets that were reviewed by the researchers.

Details commonly discussed in dataset publications include the size of the dataset, which explains the number of utterances or audio clips the dataset has

in total, and is usually represented in the Waveform Audio File format. As well as other specifications including the total number of stimuli or provocations, as well as the number of sentences uttered during the data gathering procedure. The description of the data gathering process for each dataset is also discussed in detail, as well as the emotional labels used in classifying utterances extracted for the dataset.

## 2.2 Posed Datasets

Posed datasets refer to datasets that were created in a laboratory setup and are mainly acted by professional actors in a fully controlled environment. These datasets mainly involve a set of speaker stating a certain utterance assigned to them in the proper emotional cues designated, and as such are synthetic in nature, and usually overfit the models they are trained in, resulting in models that might be able to achieve high accuracy rates with datasets of the same category, but fail to achieve high accuracy rates when tested with realistic emotions (Abbaschian et al., 2021).

As posed datasets are widely used in the field of speech emotion recognition, The Berlin Emotional Speech Database (EMODB) is stated to be the most popular choice for studies focusing on speech emotion recognition (Sun, 2020). The dataset was developed from Institute of Communication Science, Technical University of Berlin and consists of 10 sentences in 535 utterances in each of the 7 emotions in the dataset. The dataset contains utterances from 10 speakers of equal gender distribution and were cross evaluated with 20 people via human perceptions tests. The dataset was utilized in a number of studies that were synthesized, most recently from a study in 2020 that focused on the removal of feature extraction and the addition of gender as a modality (Burkhardt et al., 2005).

The same study by Sun from 2020 also included CAISA, a dataset containing 9,600 waves files of 400 utterances each evoking one of the six emotions present in the dataset. Similarly to the EMODB dataset, it was also done in a controlled environment with professional actors utterances factoring in pronunciation, text transcription with word boundaries, and Part-of-speech tagging, and featured 4 speakers of equal gender distribution, with the dataset evaluated by 10 people (Tao, Liu, Zhang, & Jia, 2008).

The use of carrier phrases in data collecting was explored in the Toronto Emotional Speech Set (TESS), from the University of Toronto. The dataset comprises a set of 200 words spoken in a phrase totaling 2800 stimuli. The layout of the



dataset includes responses to the speaker repeating a word spoken by the organizer. There were 2 actresses in total, one aged 24 years old and the other aged 64 years old (Pichora-Fuller & Dupuis, 2020).

The topic of emotions through song was explored in the The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) dataset, which was created in Canada, and is composed of 1440 speech utterances, where each utterance was vocalized twice in different emotional intensities, but also made an additional 1012 song utterances. This dataset was staged by 24 professional actors of equal gender distribution, and showcases 8 emotions (Livingstone & Russo, 2018).

CAISA is available for use in various open web sources, while EMODB can be downloaded on their website proper. RAVDESS can be collected through Zenodo and TESS can be accessed via the University of Toronto’s website.

Table 2.1: Summary of Posed Datasets

Name	Origin	Size	Description	Speaker Count	Nature of Dataset	Emotion Label
CAISA	Institute of Automation, Chinese Academy of Sciences	A total of 9,600 .wav files in 500 utterances (300 parallel texts / 200 non-parallel texts) revolving 6 emotions and a total of 12,000 sentences.	- Utterances factoring in pronunciation, text transcription with word boundaries, and Part-of-Speech tagging - Cross-evaluated by 10 people	- 2 males - 2 females	posed	- Happiness - Sadness - Anger - Surprised - Fear - Neutral
Berlin Emotional Speech Database (EMODB)	Institute of Communication Science, Technical University of Berlin	A total of 535 emotional utterances of 10 sentences in different emotions by different speakers, and a total of 800 sentences.	- Recordings taken in an anechoic chamber while using a high-quality recording equipment - Electro-glotto grams were also recorded - Data was subject to a human perception tests with 20 subjects to evaluate the quality	- 5 males - 5 females	posed	- Anger - Fear - Happiness - Sadness - Disgust - Boredom - Neutral
Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS)	Department of Psychology, Ryerson University	A total of 1440 speech utterances and an additional 1012 song utterances.	- Dataset was created by actors repeating each vocalization twice whilst being recording in two different emotional intensities.	- 12 males - 12 females	posed	- Happy - Sad - Angry - Fearful - Surprise - Disgust - Calm - Neutral
Toronto Emotional Speech Set (TESS)	Northwestern University Auditory Test No. 6, University of Toronto	A total of 200 words spoken in a phrase with 2,800 stimuli in total.	- Conducted via the conductor speaking carrier phase of Say the word followed by the designated word / phrase to the speaker.	- 2 females (24 year old and 64 year old speakers)	posed	- Happiness - Pleasant - Surprise - Sadness - Anger - Disgust - Fear - Neutral

## 2.3 Induced Datasets

Induced datasets refer to datasets that are usually created in a controlled environment with informed participants, but incorporate spontaneous or natural methods such as dyadic interaction or randomized questioning during the data gathering process in order to make the data more naturalistic (Abbaschian et al., 2021).

Creating emotional data through dyadic interaction was implemented in IEMOCAP, a dataset consisting of 51 videos of recorded dialogues, segmented into a total of 302 videos in the dataset. The use of dyadic sessions during the data gathering process of the dataset is to ensure realism in the procured data due to mutual interaction resulting in less synthetic emotional responses (Busso et al., 2008). The IEMOCAP dataset is noteworthy for being composed of both a scripted and naturalistic dataset. This is due to a portion of the dyadic sessions having the participants memorize and enact scripted utterances with a specific emotion in mind. The second subset of the dataset includes spontaneous utterances from the participants, requiring them to improvise based on given scenarios within their own terms, with the utterances being annotated thereafter (Busso et al., 2008). The IEMOCAP dataset can be accessed through the University of Southern California website.

Table 2.2: Summary of Induced Datasets

Name	Origin	Size	Description	Speaker Count	Nature of Dataset	Emotion Label
Interactive Emotional Dyadic Motion Capture (IEMOCAP)	Ming Hsieh Department of Electrical Engineering	A total 51 videos of recorded dialogues, segmented into a total of 302 videos in the dataset.	- Gathered data divided into 3-15 second utterances - Dataset was annotated and assessed by 3-4 assessors	- 10 speakers split between 5 Dyadic sessions	induced	- Neutral - Happiness - Sadness - Anger - Surprise - Fear - Disgust - Frustration - Excited - Other

## 2.4 Naturalistic Datasets

Naturalistic datasets refer to datasets that are entirely spontaneous, and are the best references of natural human emotions. They are however complicated to extract and utilize due to the datasets often having concurrent emotions being displayed together, the dynamic variation in the display of emotions, as well as having background noise present in the utterances available and the limited sources of data available (Abbaschian et al., 2021).

The collection of recordings of human interaction has been cited as a possible method of acquiring naturalistic data (Dhall, Goecke, Lucey, & Gedeon, 2012). One dataset that has utilized this method is The ‘Vera Am Mittag’ (VAM) German Audio-Visual Emotional Speech Database. This dataset contains 12-hour recordings of the German talk show of the same name. Segmented into broadcasts, dialogue acts and utterances, the database contains naturalistic emotional speech recorded from discussions among the guests of the talk show. The recordings were separated into four categories pertaining to the recording qualities: Very Good, Good, Usable and Not Usable. The ‘Very Good’ and ‘Good’ categories were

labeled as VAM I and VAM II respectively. There are 1018 samples and 47 speakers in total in VAM I and VAM II. The emotional dimensions recorded in the database are Valence, Activation, and Dominance (Grimm et al., 2008).

The prospect of recording emotional cues from unaware individuals, specifically children in this instance, led to the creation of the The Friedrich-Alexander-Universität (FAU) Aibo Dataset, which originated from the Friedrich-Alexander-University, contains 8.9 hours of speech recording of children ages 10-13 interacting with Sony’s pet robot Aibo via a close-talk microphone. The audio was automatically segmented into turns using a long pause of at least more than one second. The dataset has speakers that comprises 51 children (30 female and 20 male). It captures naturalistic speech emotion from the speakers as each word was annotated as one of eleven defined emotion states by five listeners via majority voting. The eleven defined emotion states are Neutral, Empathic, Motherese, Touchy, Reprimanding, Angry, Joyful, Bored, Hesitant, Surprised, Rest. Rest refers to emotion states that are not within the bounds of the first ten emotion states (Cao et al., 2015).

The usage of semantic analysis is utilized in the Sentiment Analysis in the Wild (SEWA), which primarily contains naturalistic recordings, one of which are audiovisual recordings from 32 pairs (64 participants). In this particular dataset, the participants were asked to watch a 90 second commercial and perform a discussion with their partners. It contains 538 annotated short recordings segments, and six different cultures were also presented (British, German, Hungarian, Greek, Serbian, and Chinese). The recordings were split into training (17 pairs), development (7 pairs), and tests (8 pairs). It captured three emotional dimensions: Valence, Liking and Arousal (Tzirakis et al., 2021).

The Vera am Mittag German audio-visual emotional speech database, can be requested in thr Sail USC website while FAU AIBO dataset can be requested through the author’s email. SEWA, which was present in a sentiment analysis paper done in 2021, can be currently accessed through requests on their website and LSSED can be requested through the author prof. Xing on behalf of an institution with the necessary end user license agreement.

A recent study in 2021 established that most models utilizing the datasets available to speech emotion recognition overfit, leaving to models that are biased towards the data they are trained on due to the relatively small size and scale of these datasets. This is rectified by the creation of The Large-Scale Speech Emotion Dataset or LSSED, which contains 147,025 different utterances. The dataset follows the discrete emotion theory, and includes the emotional labels of happy, sad, angry, disgusted, surprised, bored, fear, disappointed and neutral. An emotional label of “others” was also present in the dataset, which pertains to emotions

shown that were uncommon. The total dataset features over 200 hours of audio data, with each audio clip lasting an average of 5.05 seconds, and was conducted by having participants in an indoor lab environment with a camera pointed at them answer randomized questions, with their responses being associated with the proper emotional label by experts. The distribution of participants in the dataset include 335 male participants and 485 female participants of different age groups, which total to 820 participants. The large scale nature of the dataset is due to the assessment that the small scale nature of most datasets are often the cause of models being trained under them tending to overfit.

Table 2.3: Summary of Naturalistic Datasets

Name	Origin	Size	Description	Speaker Count	Nature of Dataset	Emotion Label
FAU Aibo Emotion Corpus	Friedrich-Alexander-Universitat	The dataset's whole corpus contains 8.9 hours of speech recording (via a close-talk microphone) in total and was automatically segmented into turns using a hecong pause of more than 1 second.	<ul style="list-style-type: none"> <li>- The dataset was originally labelled at word-level.</li> <li>- Each word was annotated as one of the eleven states by five listeners via majority voting</li> <li>- Turn-level labels were derived from word-level labels with confidence scores</li> </ul>	<ul style="list-style-type: none"> <li>- 21 males</li> <li>- 30 females</li> </ul>	naturalistic	<ul style="list-style-type: none"> <li>- Neutral</li> <li>- Empathic</li> <li>- Motherese</li> <li>- Touchy</li> <li>- Reprimanding</li> <li>- Angry</li> <li>- Joyful</li> <li>- Bored</li> <li>- Hesitant</li> <li>- Surprised</li> <li>- Rest</li> </ul>
Sentiment Analysis in the Wild (SEWA)	Imperial College London RealEyes PlayGen University of Passau	A total of 538 annotated audiovisual short recordings segments representing 6 cultures with 90 segments per culture	<ul style="list-style-type: none"> <li>- Participants were tasked in watching a 90 second commercial and discuss it with their partners</li> <li>- Three modalities were observed, namely audio, visual and text</li> </ul>	<ul style="list-style-type: none"> <li>- 64 speakers split into 32 pairs</li> </ul>	naturalistic	<ul style="list-style-type: none"> <li>- Arousal</li> <li>- Valence</li> <li>- Liking</li> </ul>
Vera am Mittag (VAM)	Communication Engineering Lab of Karlsruhe Institute of Technology, Karlsruhe, Germany	Taken from 12 hours of recording from a show titled, Vera am Mittag totaling 1018 samples split into two parts, namely VAM I and VAM II	<ul style="list-style-type: none"> <li>- Speakers classified based on readability of recording, with classifications being Very Good, Good, Usable and not Usable</li> <li>- Only speakers classified as Very Good (VAM I) and Good (VAM II) were used</li> </ul>	<ul style="list-style-type: none"> <li>- 19 speakers for VAM I</li> <li>- 28 speakers for VAM II</li> </ul>	naturalistic	<ul style="list-style-type: none"> <li>- Valence</li> <li>- Activation</li> <li>- Dominance</li> </ul>
The Large-Scale Speech Emotion Dataset (LSEED)	South China University of Technology	A total of 147,025 utterances across 9 different emotional labels amounting to over 200 hours worth of data	<ul style="list-style-type: none"> <li>- Participants were subject to an indoor laboratory environment</li> <li>- Were tasked with answering random questions</li> <li>- Participants had a camera pointed at them</li> <li>- Answers and responses were appropriately labeled by an expert team Consists of 820 total speakers with a gender ratio of 335 Males and 485 Females</li> </ul>	<ul style="list-style-type: none"> <li>- 64 speakers split into 32 pairs</li> </ul>	naturalistic	<ul style="list-style-type: none"> <li>- Neutral</li> <li>- Happy</li> <li>- Sad</li> <li>- Angry</li> <li>- Fear</li> <li>- Surprise</li> <li>- Disappoint</li> <li>- Disgust</li> <li>- Other</li> </ul>

## 2.5 Datasets and Emotion Models

The previous section presents the various datasets based on the nature of the data collected. This section discusses the various methods used to create emotion models using these datasets, including its performances measured based on accuracy scores.

Regarded as the standard of speech emotion recognition (Sun, 2020; Ntalampiras, 2021), EMODB has an accuracy rate being at a high 80-90% rate depending on the model or algorithm used based on the unweighted accuracy rate, while also being able to boast a high usage rate in accordance to the number of researches that have used the aforementioned dataset. Similarly, CAISA also boasts a high accuracy rate that ranges from 79% to a highest rate of 84%, although the dataset is less common in practice than EMODB and is therefore not as well documented or proven at the time of writing.

In a comparative study conducted in 2018, two datasets were utilized for the purposes of the study, with RAVDESS being used to train the data, and TES to test the data. The values seen in the RAVDESS portion of the table pertain to the validation results from the analysis, with the model both being trained and tested with RAVDESS. RAVDESS marks a 56-70% accuracy rate, with a model having a 90% accuracy rate, although it should be taken into account that the model only uses 2 classes, with the same model with 12 additional classes having a 20% accuracy loss (resulting in 70% accuracy). TESS has a lower accuracy rate, having a peak of 66%, and a low of 31.5% accuracy rate. The breakdown of these results are presented in Appendix A.

The usage of induced datasets in recent years has also been explored, with the IEMOCAP dataset being the only one to appear during the literature review. While as popular in usage as the EMODB dataset in the field of speech emotion recognition, IEMOCAP yielded a far lower performance in terms of accuracy, ranging only from 54% when used in CTC-BLSTM to 71% when using the ASR transfer learning method in terms of accuracy rating. The lower rating of the IEMOCAP dataset may be due to its classification as an induced dataset, rather than the fully posed or acted nature of the datasets listed above it.

Older naturalistic datasets such as the VAM dataset, utilized recordings from a German TV talk show named “Vera Mittag”. The VAM dataset appeared only in one instance in literature review wherein the results stated in the literature were compared against the different variations of the VAM dataset, namely VAM I and VAM II. The first variation which was performed with VAM I yielded the highest results ranging from 75% to 78% in correlation of ten-fold cross validation, while VAM II, the second variation, yielded the lowest results ranging from 49% to 63%. The third variation which utilizes both versions of the VAM dataset (VAM I + VAM II) yielded relatively poor results compared to the first variation, but slightly better from the second variation ranging from 62% to 68%.

On the other hand, a comparative study in 2021 tackled various speech emotion recognition methods on different speech emotion databases using FAU AIBO, with the dataset from Germany being utilized in a study to observe the performance of ranking SVM, and it yielded a 44.4% recognition rate. On the other hand, Speech emotion recognition methods: A literature review (2017) tackled deep learning techniques for speech emotion recognition (Abbaschian et al., 2021). The study used the FAU Aibo dataset to study the performance of three different deep learning algorithms. The first method was LSTM with Multitask Learning (ML) which yielded an average accuracy of 52%. The second method was LSTM with Generative Adversarial Network (GAN), and it yielded an average accuracy of 64.86% which was the highest among the three methods. The last method was Linear Discriminant Analysis (LDA) with Transfer Linear Subspace Learning (TLSSL)

and it yielded an average accuracy of 54.61%. Table 2.2 shows the summary of algorithms used to model emotion datasets.

The use of semantic primes was utilized in the SEWA dataset which was used in a semantic analysis paper in 2021, where the dataset was utilized in training and testing the model and verifying the projected weights of each emotion in the dataset in comparison to the actual results (Tzirakis et al., 2021). The SEWA dataset deviates from discrete emotions, using semantic primes instead in its formulation.

A recent study in 2021 established that most models utilizing the datasets available to speech emotion recognition overfit, leaving to models that are biased towards the data they are trained on due to the relatively small size and scale of these datasets. This is rectified by the creation of The Large-Scale Speech Emotion Dataset or LSSED, which contains 147,025 different utterances. The dataset follows the discrete emotion theory, and includes the emotional labels of happy, sad, angry, disgusted, surprised, bored, fear, disappointed and neutral. An emotional label of “others” was also present in the dataset, which pertains to emotions shown that were uncommon. The total dataset features over 200 hours of audio data, with each audio clip lasting an average of 5.05 seconds, and was conducted by having participants in an indoor lab environment with a camera pointed at them answer randomized questions, with their responses being associated with the proper emotional label by experts. The distribution of participants in the dataset include 335 male participants and 485 female participants of different age groups, which total to 820 participants. The large scale nature of the dataset is due to the assessment that the small scale nature of most datasets are often the cause of models being trained under them tending to overfit.

## **2.6 Machine Learning in Speech Emotion Recognition**

In this section, the research papers were labeled and categorized based on the algorithm present in the models. This section includes the various inputs used in each algorithm, the problems that these algorithms intend to solve, as well as the weaknesses, model description, results and their possible improvements from the models

A review of papers on speech emotion recognition showed the prevalent use of the end-to-end model. To discuss further, end-to-end speech emotion recognition is based on the end-to-end speech recognition model wherein its structure

is greatly simplified into a single model compared to a conventional pipeline of speech recognition systems (Sainath, 2020).

Some literature discusses and tackles End-to-End models focusing on the features of Gender and Age information from speech datasets. Additionally, the latest literature dwells on a new algorithm or approach, most notably the Semantic Analysis, which further looks into the interpretation of text.

One study in 2018 utilizes the End-to-End Multimodal Model algorithm. The algorithm used, a convolutional recurrent neural network, utilizes IEMOCAP as the dataset for extraction, with the main inputs being Speech Network and Deep Residual Network handling each respectively. The audio and video data was concatenated into a 2-layer Long short-term memory. An additional modality of Gender Information was also added into the model. The study aimed to address the problems of the usage of static weights adversely affecting the model performance as well as the problems of segregating features and data for usage and has stated the weakness of the chosen algorithm to be ineffective when using additional modalities. Further improvements as stated by the article were stated to be possible usage of other modalities could be explored, with modalities such as age being a worthy option to take into account (Chae et al., 2018).

Similarly, a study conducted in 2019 conducted a comparison of various algorithms, namely convolutional neural networks, Hidden markov models and Long Term Short Memory, to evaluate the best audio feature and best model architecture. RAVDESS is used for training the data while TESS is used for testing the data with all models using Audio via Time-domain features from the Log-Mel Spectrograms and Frequency-domain features extracted via Mel Frequency Cepstral Coefficients (MFCC), as well as an additional modality of Gender Information being included in some models. The results show using unweighted accuracy on different algorithms and was intended to solve the problems related to the difficulty of Finding the best audio feature and model architecture but states the weaknesses of the algorithms to be its features being more dependent than the model architecture, the algorithm’s habit of being confused on gender-specific emotions as well as its limit to working on the same language. Possible improvements include the usage of Atrous Spatial Pyramid Pooling (ASPP) to learn features better as well as a hierarchical structure for gender and age-groups for improvement in performance(Venkataramanan, Rajamohan, 2019) which in the case of gender information, was proven to be effective at increasing accuracy in a study conducted a year prior in 2018 (Chae et al., 2018) and a more recent study conducted in 2020 that illustrates the increased accuracy rates when a gender information block was added (Sun, 2020).

With the concerns on the lack of datasets in the field, as well as the heavy

computation complexity problems present in the field, a study was conducted in 2020, using Mel Frequency Cepstral Coefficients (MFCC) to create the model. The resulting model revealed many shortcomings, including the untested nature of the model, as well as its confusing neutrality with sadness, its limited in frame-level prediction and its ignorance of sequential information in emotion labels decoding. Further study as explained in the article include the addition of another modality, specifically a semantic feature (Zhou & Beigi, 2020).

With the literature synthesized utilizing different forms of feature extraction, one study conducted in 2020 discusses the difficulty and tedious nature of collecting data and feature extraction, and as such aimed to remove that particular process through the use of the R-CNN algorithm inside an end-to-end model. As such, the model was able to automatically extract features from an input consisting of raw data, removing the need of manual feature extraction. The model had an additional modality, being gender information, with the model extracting data from 3 datasets, namely CAISA, EMODB and IEMOCAP. While the model was able to bypass manual feature extraction, the model produced had shortcomings including its inability to work in real time, as well as its steep computational expense. Recommendations for the improvement of the algorithm include exploration on the prospect of the algorithm to be able to work in real time, as well as the possibility of other modalities being looked upon without having the need to meticulously classify speech data. The possibility of reducing computational expenses can also be taken into account (Sun, 2020).

The issue of feature extraction was addressed in a different light in a recent study conducted in 2021, utilizing a Siamese Neural Network with the features being the audio via Log-Mel Spectrograms and Temporal Modulation and EMODB being the dataset used for extraction. The study aimed to address the issue of most models relying on features to classify the right emotion as well as the need to feed massive quantities of labeled data to get the best accuracy. The resulting model was able to provide the most accurate prediction when using a small training dataset with an average recognition rate of 82.1% but is subject to being prone to misclassification as well as having a generally weak performance. Further improvements such as adding more classes in the dictionary to find similarities and dissimilarities better as well as investigation on sufficient conditions in the training set and adding more quantity in the training set can be expounded upon (Ntalampiras, 2021).

A new method of feature extraction was explored in a study in 2021, discussing the proposal of a method that comprises two networks: a semantic feature extractor and a paralinguistic feature extractor. The semantic feature extractor extracts high level features containing semantic information of the input through Speech2Vec and Word2Vec. The paralinguistic feature extractor extracts the low



level features containing paralinguistic information of the signal through a network that comprises three 1-D CNN layers with a rectified linear unit (ReLU) as activation function. Both semantic and paralinguistic feature vectors are then passed through a fusion layer and are passed through a LSTM model for final prediction. The model was able to get the best results in valence and liking dimension (0.503  $\rho c$  and 0.312  $\rho c$  respectively), and second best result in arousal dimension with 0.583  $\rho c$ . The objective function used is the Concordance Correlation Coefficient ( $\rho c$ ) which was also used in the AVEC 2017 Challenge papers that the model was compared with. The SEWA dataset was used in the study, with deficiencies in the resulting study being the necessity to use two different networks to separately capture semantic and paralinguistic features as well as lack of performance testing on categorical emotion recognition datasets. The study suggested applying the principle of the multimodal framework in a single end-to-end model to help in terms of model simplicity (Tzirakis et al., 2021).

Common issues present in each study focus many on feature extraction from datasets and the requirement of large training data for accurate results. Neural networks were utilized as the preferred algorithm in almost all of the models reviewed, with Convolutional Neural Network (CNN), Time Delay Neural Network (TDNN) and Siamese Neural Network being examples of the application of Neural Network to models of speech emotion recognition. The outlier in the models synthesized was a model that evaluates using the three emotional dimensions, namely arousal, valence and liking, that are captured in the SEWA dataset. The Concordance Correlation Coefficient ( $\rho c$ ) was used as its objective function.

Furthermore, the usage of EMODB was cited in many articles present during the literature review, with one study in particular stating EMODB to be one of the standard dataset in the field of speech emotion recognition (Sun, 2020). In extracting features from these datasets, MFCC and Long- Mel Spectrograms, with Audio as the main feature, are the most frequently used between different algorithms. Some models have also added R-CNN into their model, removing the need to manually extract features from the data. More recent studies, specifically those that utilized naturalistic datasets, have used feature vectors, both semantic and paralinguistic as main features, differentiating from the other algorithms mentioned above.

The usage of both CNN and LSTM in conjunction was conducted in a study in 2016, wherein a CNN pooling was used to preprocess the features, but the recurrent layers in the model utilized two bidirectional LSTM. This eliminates the need of feature extraction, and allows the model to use raw data signals as input. The model was tested on the RECOLA dataset, which contains 5 minut utterances from 46 individuals (Trigeorgis et al., 2016).

A breakdown of the different models and algorithms implemented by the studies stated above may be viewed in Appendix A.

From the literature review, it can be concluded that the field of speech emotion recognition has improved with the introduction of CNN, which has been prevalent in its usage in the mentioned field. The topic of feature extraction and its difficulty has been stated between many articles, most notably in End-to-End speech emotion recognition With Gender Information (2020), which has tackled the issue, resulting in its use of Region Based Convolutional Neural Networks, which removed the need to manually extract necessary features from the data, but resulted in a computationally expensive model that cannot operate in real time. Further research on this can be done in order to eliminate the need to manually extract features, which removes the difficulty of being able to produce the necessary data compatible with the models. The usage of additional features and modalities has also been expressed in the various literature, with age being a prospect for future recommendations (Sun, 2020).

# Chapter 3

## Theoretical Framework

### 3.1 Emotion Theory

In the field of emotion theory, the origin of emotion and its emergence has been subject to speculation and study between various scientists and theorists alike, in regards to the current research accumulated, there are three widely accepted and followed approaches to understanding emotions and the behaviors and patterns surrounding them.

#### 3.1.1 Categorical Approach

Emotions may be labeled using discrete labels based on the categorical approach. The approach describes emotions to be comprised of a unique set of semantic primes that are all relatively fixed in regards to trigger patterns, how it is expressed as well as personal experiences to the emotion. The basic emotions are commonly seen in these theories often include joy, fear, anger, sadness, disgust, and anger (Jeon, 2017). The details and number of basic emotions present often differ from one study to the next, with many well-known personality theorists and psychologists having their definition and set of basic emotions.

Silvan Tomskins states that there are only 9 affect states, with these states are present in every human being and are paired, with each affect pair about the least and most intensive expression of that effect. The display of these emotions to others garner the same response from those individuals, with these phenomena being termed as affective resonance, or more commonly known as empathy,

which influences the overall intent of the individual, translating towards a certain emotion (Nathanson & Lansky, 1997).



Figure 3.1: Visualization of Ekman’s Theory (Ekman, 2021)

Ekman’s theory on the categorical approach remains to be the most universally accepted stance and is the most common adaptation of the approach in the field of speech emotion recognition. Based on the experiments on static photography of facial expressions, he and his team concluded that there are 6 basic emotions present, namely happiness, fear, surprise, sadness, disgust, and anger (Gunes & Pantic, 2010).

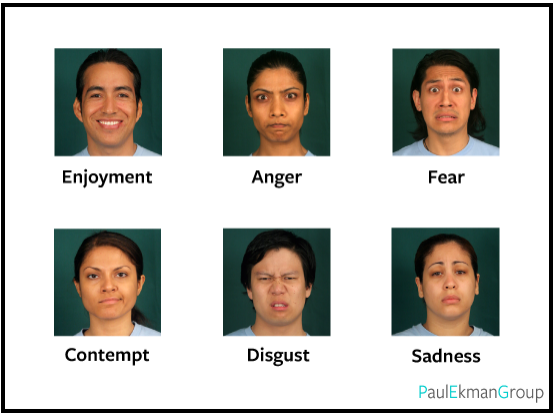


Figure 3.2: Visualization of the Six Basic Emotions (Ekman, 2021)

Basic emotions are defined as a limited set of emotions with unique trigger patterns and behavioral properties (Ekman, 1992b; Russell, 2006) as well as be-

ing continuously innately developed as a response to ubiquitous regular-spanning events. Each of these emotions has innate nervous substrates and common behavioral characteristics within each emotion (Shpigler et al., 2017) and are the bases that form the construction of more complex emotions, with these complex emotions sharing elements from multiple basic emotions (Ekman, 1992a).

These emotions are hypothesized to be linked to the essentials of a living being, as an indicator of the being’s innate needs (Schoeller et al., 2019). This is expounded in a famous theory by Grossberg and Levine in 1987 that states basic emotions to be neural signals that connect to the instinctual sensors of the brain, making the instinctual needs of the individual to be more pronounced and more likely to be noticed (Grossberg & Levine, 1987). This characteristic of basic emotions is the indicator that sets itself apart from complex emotions, which are more connected to the higher needs of the human being (Perlovsky, 2012).































Upper Face Action Units					
AU1	AU2	AU4	AU5	AU6	AU7
					
Inner Brow Raiser	Outer Brow Raiser	Brow Lowerer	Upper Lid Raiser	Cheek Raiser	Lid Tightener
*AU41	*AU42	*AU43	AU44	AU45	AU46
					
Lip Droop	Slit	Eyes Closed	Squint	Blink	Wink
Lower Face Action Units					
AU9	AU10	AU11	AU12	AU13	AU14
					
Nose Wrinkler	Upper Lip Raiser	Nasolabial Deepener	Lip Corner Puller	Cheek Puffer	Dimpler
AU15	AU16	AU17	AU18	AU20	AU22
					
Lip Corner Depressor	Lower Lip Depressor	Chin Raiser	Lip Puckerer	Lip Stretcher	Lip Funneler
AU23	AU24	*AU25	*AU26	*AU27	AU28
					
Lip Tightener	Lip Pressor	Lips Parts	Jaw Drop	Mouth Stretch	Lip Suck

Figure 3.3: Table of Action Units for Upper and Lower facial expressions (De la Torre et al., 2015)

To distinguish each emotion based on expressions, the Facial Action Coding System (FACS) was published by Ekman and Friesen as a method used in reading emotions through the facial expressions of individuals. This system maps the emotion present in an individual’s expression through action units(AU), which outwardly noticeable facial movements that are each tagged as a unique action unit which is each scored based on a 5-point intensity scale (Clark et al., 2020).

Basic expressions	Involved Action Units
Surprise	AU 1, 2, 5, 15, 16, 20, 26
Fear	AU 1, 2, 4, 5, 15, 20, 26
Disgust	AU 2, 4, 9, 15, 17
Anger	AU 2, 4, 7, 9, 10, 20, 26
Happiness	AU 1, 6, 12, 14
Sadness	AU 1, 4, 15, 23

Figure 3.4: Table of AU subsets for each basic emotion (Ghayoumi, Bansal, 2016)

A total of 66 AUs have been created from various facial movements to map out the 6 individual basic emotions based on Ekman’s theory, with each emotion having a subset of AUs that are needed to display the targeted emotion (Ghayoumi & Bansal, 2016).

Caroll Izard states that discrete emotion feelings cannot be created, taught, or learned via cognitive processes and that perceptual and conceptual processes and consciousness itself are effects of emotions rather than sources of their origin (Jeon, 2017). Furthermore, discrete emotion experiences emerge in early development well before children acquire language or the conceptual structures that adequately frame the qualia known as discrete emotion feelings (Izard, 2009).

Several psychologists disprove this, however, stating that the display of cognitive mental states occurs more often than the basic emotions often described in the categorical approach, prompting the necessity to go beyond the range of the basic emotions commonly cited, as it is evident that these emotions are not able to cover all the common states present in a human’s mental state (Gunes & Pantic, 2010).

### 3.1.2 Dimensional Approach

Emotions are classified as an internal state and therefore are human behavior (Baker, 2004) but rather as the state that induces such behavior. As such, behaviors that are linked to or are induced by an emotion commonly have 2 features (LeDoux & Brown, 2017), with the vertical dimension housing behavior agitation, and the horizontal dimension deciding the approach-avoidance psychological conflict.

The dimensional approach states that all emotional states are linked in a neurophysiological system, which is responsible for triggering all of them. This places the theory in stark contrast to the discrete emotion theory, as the dimensional

models explain all emotions as being linked together. Emotions in this context are often explicated to be in at least two or three dimensions, with studies under this approach using valence and arousal, and/or incorporate domination as its dimensions (POSNER, RUSSELL, & PETERSON, 2005). Arousal as a dimensional feature pertains to the intensity of involuntary activation that an event or experience generates, with low arousal states being linked to calmness, and high arousal states being connected to the excitement, while valence is the negative to the positive scale of the affability generated from an event or experience (Bestelmeyer, Kotz, & Belin, 2017).

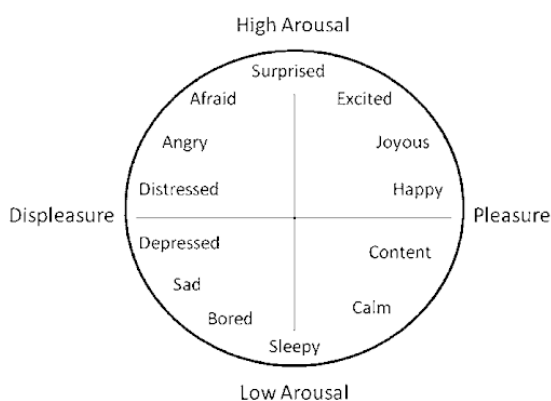


Figure 3.5: Circumplex model (Ekman, 2021)

James Russell states in his circumplex model that all emotions are encompassed in a two-dimensional circular graph with two dimensions, namely valence, and arousal, represented by the vertical and horizontal axis respectively. The emotional state shown would be based on the intensity of both valence and arousal in the graph, with each emotion having its certain value of both valence and arousal (Rubin & Talarico, 2009).

A vector model incorporating valence and arousal as its two dimensions were also formed, which states that an underlying arousal dimension would always have a valence dimension whose direction would signify which emotion is being shown, with positive emotions having high valence dimensions, and negative or neutral emotions having low valence. Arousal typically plays a higher role in characterizing emotions with high valence states, but is less prominent in emotions with low valence states (Rubin & Talarico, 2009).

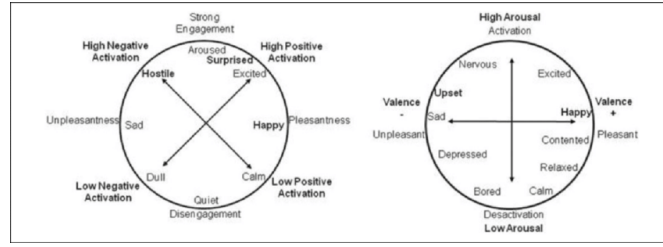


Figure 3.6: Positive activation negative activation model (Reyes et al, 2019)

Similarly, a study in 1985 also formulated the Positive activation negative activation model, in which emotions characterized with high arousal are then characterized by their valence value, and are much less prominent with emotions with low arousal states, which operates similarly to the vector model (Watson & Tellegen, 1985).

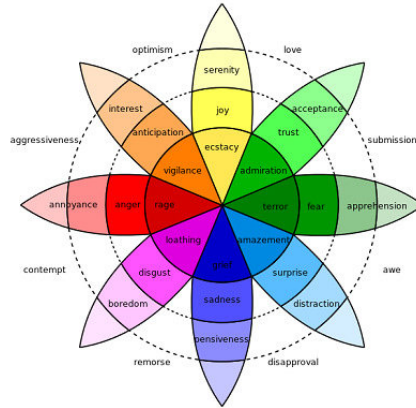


Figure 3.7: Wheel of emotions model (Six Seconds, 2021)

Robert Plutchik on the other hand states that there are 8 basic emotions, each with a low, medium, and high intensity. These 8 basic emotions were based on adaptive biological mechanisms (Plutchik & Kellerman, 1980). These were categorized due to physiological purpose, similarly to a cone-shaped model, commonly known as The Feelings Wheel (*Plutchik's Wheel of Emotions: Feelings Wheel Six Seconds*, 2021). Plutchik's model both incorporates elements from the dimensional models of emotion, with the emotions were placement being based on a person's psychological purpose, but also has elements from the studies under the categorical approach, in which Plutchik states there to be 8 basic emotions each with 3 varying degrees of intensity.

The dimensional approach is argued, however, most notably by discrete emo-



tion theorists that the minimization of emotions into two or three dimensions results in a large loss of features and information. In addition, some emotions also become indistinguishable under this approach, such as fear and anger (Gunes & Pantic, 2010).

### 3.1.3 Appraisal Approach

The subject of continuous evaluation on emotion responses is taken into primary consideration in the appraisal approach, which states that emotions are brought about through a continuous evaluation of both the internal state of the individual, as well as the external state of their environment. This further elaborates that emotion comes from multiple stimulation checks and that emotions from this approach are not limited to a set number of emotions, but come from various emotional states stemming from distinct appraisal patterns. While few to none rival this emotion theory, the application of the appraisal theory in automatic emotion recognition is subject to further study and research, as the complexity of the measurements of change still cannot be fully conducted through machines (Gunes & Pantic, 2010).

The appraisal approach and its connection to stress were proposed by Richard Lazarus in his book entitled *Psychological Stress and Coping Process* in 1966 in which he stated that stress is the variance between a person and their ability to cope or not. The reaction to stress and the outcome differs from one individual to the next, which is determined by their response and interpretation towards a certain event and the usual mindset that comes in effect from it, termed as appraisals (Lazarus & Folkman, 1984).

A further discussion and hypothesis of emotion being defined by four features is discussed by Sander and Scherer states in their book entitled *Oxford Companion Emotion and Affective Sciences* in 2009 that state that emotion is defined using 4 fundamental features that distinguish one affective state from another. These features are namely a person's mood, preference, attitude, as well as traits and personal mental well-being states (Sander & Scherer, 2009).

## 3.2 Signal Processing

This section of the paper will discuss the pre-processing of speech via audio files before feeding them into the model. However, the use of raw data can be limited in Machine Learning since there is a probability that the quality of the data is

bad. Speech in itself has information, also known as features, that is accessible and extracted for machine learning or analytics (Chaudhary, 2020). Once all the information is available from the data, it can be used for the model.

### 3.2.1 Speech in Emotion Recognition

Speech has been the primary medium in human interaction. Speech conveys information that contains emotion cues as a form of communication between humans (Ntalampiras, 2021). As a result, speech can be a determining asset in assessing the emotions portrayed by an individual (Breazeal & Aryananda, 2002). Because of this, speech is a primary modality in determining the emotions present when it comes to modeling emotion recognition in machines (Breazeal & Aryananda, 2002). Additionally, features can be extracted from audio, which can be used to effectively recognize emotions. Common features that are extracted are prosodic features such as pitch and energy. Features are extracted through the use of speech analysis methods, which have proven effective in recent times (Neiberg, Elenius, Karlsson, & Laskowski, 2006).

### 3.2.2 Fourier Transform

Audio signals are complex signals that comprise a series of single-frequency sound waves, which travel as a disturbance in the medium (Chaudhary, 2020). The resultant amplitudes are only captured when sound is being recorded. Fourier Transform (**FT**) is responsible for splitting an audio signal into a series of pure frequencies. The Fourier Transform is used in transforming the domain of a signal from time to frequency (Maklin, 2019). However, not only the frequencies are present resulting in the Fourier Transform, but also the magnitude of each of the present frequencies found in the audio signal (Chaudhary, 2020).

#### Fast Fourier Transform

$$x[k] = \sum_{n=0}^{N-1} x[n] e^{\frac{-j2\pi kn}{N}} \quad (3.1)$$

The Fast Fourier Transform (**FFT**) is a mathematical process that calculates the Discrete Fourier Transform (**DFT**) (Chaudhary, 2020). As implied by the name, the FFT cuts the DFT from  $O(n^2)$  to  $O(N \log_2 N)$ . Discrete Fourier

Transform has the same concept as Fourier Transform, which has been used in applications such as signal processing, data analysis, and machine learning algorithms (Yu, Maddah-Ali, & Avestimehr, 2017). The difference between a Discrete Fourier Transform and Fourier Transform is the input of the former is a discrete signal while the latter is a continuous signal.

From Equation 3.1,  $x[n]$  pertains to the discrete signal, where  $N$  signifies the domain size, multiplied each of the value by a value of  $e$  raised to a function of  $n$  (Maklin, 2019). In the context of algorithms, it will take  $N$  multiplication times  $N$  additions, which results to  $O(n^2)$  (Maklin, 2019).

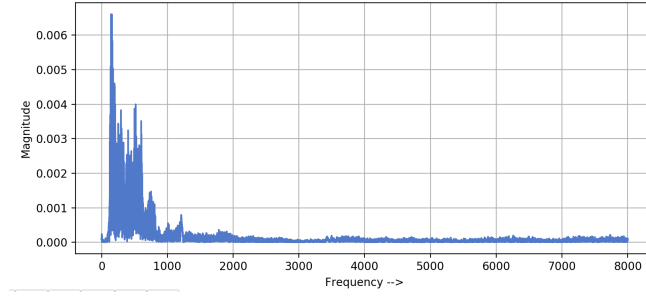


Figure 3.8: Visualization of a frequency-domain representation of a signal (Chaudhary, 2020)

Based on Figure 3.8, it is possible to find the magnitude from a frequency. However, the time information of an audio signal is lost once the Fast Fourier Transform is applied. This is where the application of the spectrogram becomes important.

### Short Time Fourier Transform

The Short Time Fourier Transform or STFT is a representation of a signal in the time-frequency domain, and this is computed by getting the discrete Fourier transforms or DFT over short overlapping windows (McFee et al., 2015). This means that when the signal is divided into shorter segments of equal length, STFT computes the DFT separately on each segment through overlapping series of windows.

$$\begin{aligned}
 X_m(\omega) &= \sum_{n=-\infty}^{\infty} x(n)w(n - mR)e^{-j\omega n} \\
 &= DTFT_{\omega}(x \cdot SHIFT_{mR}(w))
 \end{aligned} \tag{3.2}$$

To explain the formula for STFT, first is that  $x(n)$  refers to the input signal at time  $n$ . Then the  $w(n)$  refers to the length of M window function. The most common window function that is used for STFT is the Hann window function (Allen & Rabiner, 1977).  $X_m(\omega)$  pertains to the DFT of the windowed data centered around time  $mR$ . Finally,  $R$  refers to the hop size or length between successive DFTs (Allen & Rabiner, 1977).

### Hann Function

Hann function is named after the Austrian meteorologist Julius von Hann, where it is more known as a window function that performs hann smoothing. Windowing function is normally done when we would want to obtain the right timing to cut the specific signal and obtain the subset of the sample without any leakage, in the hann window you can observe that the two sides both ended in 0 making the start point and end pint equal to each other, thus lessen the spectral leakage. (Virtanen et al., 2020)

$$w(n) = 0.5 - 0.5 \cos\left(\frac{2\pi n}{M-1}\right) \quad 0 \leq n \leq M-1 \quad (3.3)$$

The function that is used in the experiment returns the window with the maximum value normalized to 1, where  $n$  means the current segment the point is at, and  $M$  means the number of points in the output window (Virtanen et al., 2020)

### 3.2.3 Discrete Cosine Transform

The Discrete Cosine Transform or DCT was first proposed by Nasir Ahmed in 1972. It is a widely used transformation technique in digital signal processing as well as data compression (“Discrete Cosine Transform (DCT)”, 2006). It is a technique that converts a signal into frequency components, wherein the input signal is represented as a linear combination of data (Virtanen et al., 2020).

$$y_k = 2 \sum_{n=0}^{N-1} x_n \cos\left(\frac{\pi k(2n+1)}{2N}\right) \quad (3.4)$$

The formula is the computation of the DCT Type 2, where  $N$  refers to the size of the matrix or array. The DCT Type 2 transform is an equivalent (up to

a scale-factor of 2) to a Discrete Fourier Transform or DFT of  $4N$  inputs of even symmetry with some elements being zero, which is the even-indexed elements (Virtanen et al., 2020).

$$f = \begin{cases} \sqrt{\frac{1}{4N}} & \text{if } k = 0 \\ \sqrt{\frac{1}{2N}} & \text{otherwise} \end{cases}$$

The resulting  $yk$  is multiplied by a scaling factor  $f$ . This additional operation for the value in conjunction to computing the transform matrix in DCT Type 2 equation is used so that the resulting matrix of coefficients would become orthonormal (Virtanen et al., 2020).

### 3.2.4 MFCC

Mel-frequency cepstral coefficients are a collective that make up a Mel-frequency cepstral (MFC). They form a cepstral representation of a audio signal with the frequency banded in the mel spectrum spaced evenly in order to more closely emulates a human's hearing (Nair, 2018).

$$C(x(t)) = F^{-1}[\log(F[x(t)])]$$

Figure 3.9: Cepstrum Conversion Formula (Elelu, 2021)

The process of extracting the MFCCs from an audio signal usually involve taking the time signal of the audio signal and applying Fourier Transform on it. This results in the spectrum of the audio signal in which the logarithmic value of magnitude of the spectrum is taken. Afterwards, the resulting value's spectrum is then taken through the use of a cosine transform, which results in the cepstrum, with the coefficients being the amplitudes of the resulting cepstrum (Nair, 2018).

### 3.2.5 Spectrogram

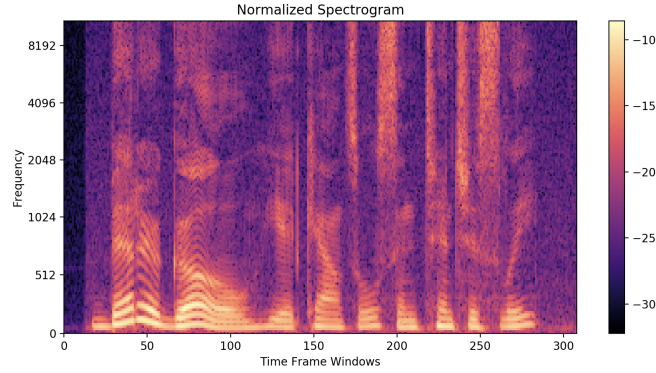


Figure 3.10: Visualization of a spectrogram from a given time frame (Chaudhary, 2020)

A spectrogram is a visual representation of pure frequencies of a given signal concerning time (Chaudhary, 2020). The domain corresponds to time and the range corresponds to the range of frequencies in Hertz ( $Hz$ ). The colors represent the magnitude of a frequency at a specific time.

The ideal use of a spectrogram is to break the entire audio signal into multiple frames that is 20ms to 30ms for each frame. It is possible to find the frequencies using the DFT or FFT for all the frames. Each frame, known as window, will represent time. For the best result, all frames must be overlapping to avoid the potential loss of frequencies.

It is now possible to create a spectrogram, which is specifically a 2D matrix of frequency magnitudes concerning time for an audio signal. Depending on which output, whether the matrix itself or the image of a given spectrogram, it is possible to feed the data into a deep learning model and it is now possible to create a model for speech emotion recognition.

The formula of the mel-scale is:

$$m = 1127 * \log\left(1 + \frac{f}{700}\right) \quad (3.5)$$

A mel spectrogram is a visual representation similar to a spectrogram with the difference being that the frequencies are now converted to the Mel scale. The domain corresponds to Time and the range corresponds to the Mel scale. The colors represent the magnitude of a frequency in the Mel scale at a specific time.

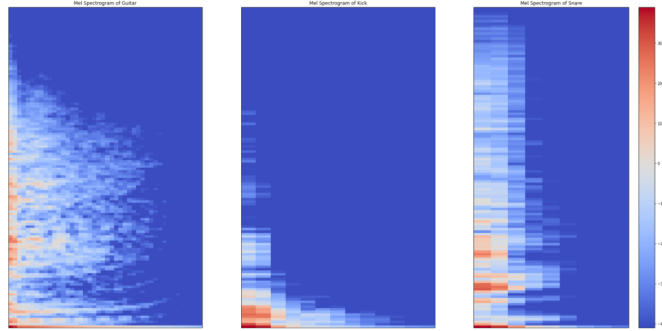


Figure 3.11: Visualization of three sample mel spectrogram (Sabra, 2021)

### 3.3 End-to-End Model Framework

The model in the study will utilize the end-to-end framework. The end-to-end framework merges the various layers into a single model, which simplifies the process (Wang & Li, 2019).

The traditional approach of a model is composed of different layers. In the context of speech recognition, an example of a traditional pipeline consists, in order, the input, feature extraction, phoneme detection, word composition, concerning and text transcript (Roza, 2019). Training under the conventional approach is time-consuming (Wang & Li, 2019). To improve the performance or the accuracy of the traditional model, each of the layers in the pipeline must be optimized under different criteria since each layer has its function.

In the end-to-end approach, the whole pipeline is replaced with a neural network (Roza, 2019). This allows only one criterion for optimization instead of different criteria from the traditional approach. The end-to-end approach simplifies the complexity since there is no need to manually label information as the neural network is responsible for learning on its (Wang & Li, 2019). The network can directly convert the input signal into corresponding mapped output, which bypasses the intermediate step in traditional algorithms (Sun, 2020).

### 3.4 Convolutional Neural Networks

The usage of convolutional neural networks has seen a surge in popularity within the field of speech emotion recognition. Recent studies utilizing CNN in one form or another has yield promising results in overall model efficacy. Studies that use

CNN often yield high accuracy rating in the models they are incorporated in, such as the model made by Sun which performed exceptionally well when compared to other deep learning models in all aspects (Sun, 2020). Other studies include one that tackled semantic analysis, resulting in a model that was able to get the best results in the dimensional features of valence and liking (Tzirakis et al., 2017).

The model in the study will utilize the Convolutional Neural Networks (CNN). CNN is composed of multiple layers of artificial neurons. These artificial neurons are mathematical functions that calculate the weighted sum of multiple inputs and outputs as an activation value (Albawi, Mohammed, & Al-Zawi, 2017).

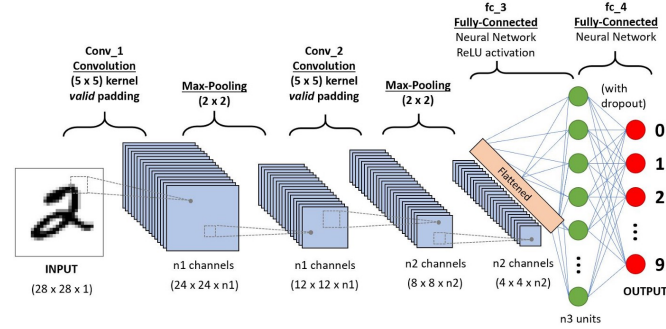


Figure 3.12: Convolutional Neural Network Architecture (Saha, 2018)

CNN contains the input, hidden and output layer. The data gets into the CNN through the input layer and passes through the various hidden layers before entering the output layer (Anwar, 2021).

### 3.4.1 Hidden Layer

The hidden layers in the network provide a basic building block to transform the data. The data is passed through various hidden layers before it is finally presented to the output layer.

#### Layer Function

Fully connected layers consist of linear functions between the input and the output (Anwar, 2021). In contrast, convolutional layers are applied to 2D and 3D input feature maps. The trainable weights are a 2D or 3D kernel or filter that moves



across the input feature map that generates dot products with the overlapping region of the input feature map (Anwar, 2021).

## Convolutional Layer

Convolutional layers are tasked with performing feature extraction from a certain input, and are mainly focused on filtering input data to be used for classification (Stewart, 2019).

$$(P * K * S + 1) * filter \quad (3.6)$$

There are three parameters in defining a convolutional layer. The first parameter is the kernel size (**K**). It pertains to the size of the sliding kernel or filter. The second parameter is the stride length (**S**). It defines how much the kernel slid before the dot product is carried out to generate the output pixel. The third parameter is the padding (**P**). It is the frame size of zeros inserted around the input feature map.

## Stride

Stride pertains to the shift size over the input matrix and is the marker of how the convolution process moves along the x and y axis of the input matrix. Stride has two dimensions, mainly the x and y axis, and has a default value of (1,1), which indicates a single shift towards the x axis and the y axis at every iteration (Brownlee, 2019a).

## Padding

Padding is the process of adding zeroes to the edges of an image to both prevent shrinkage, as well as bringing pixels at the border closer to the middle, increasing their contribution and increasing the utilization of pixels at the borders, which are often underused in comparison to pixels placed nearer to the middle of the image.

$$P = \frac{S \frac{I}{S} - I + F - S}{2} \quad (3.7)$$

The padding equation used in the model has 3 variables, the **I** or input variable signifying the input data to apply to pad too, the **F** variable that represents the filter size, and the stride variable represented by the **S** variable. The resulting image will have dimensions equal to the product of the filter size and the stride.

## Pooling Layer

The pooling layer is utilized to change the spatial size of the input layer, depending on the type of the pooling method. Max pooling decreases the spatial size of the input layer based on selecting the maximum value in a receptive field defined by the kernel. Average pooling does the same behavior by selecting the average value in a receptive field instead of the maximum value (Anwar, 2021).

$$x = \frac{input\_shape - pool\_size + 1}{pool\_size} \quad (3.8)$$

The formula for max-pooling uses 2 variables wherein the **input\_shape** is defined as the dimensions of the total input dataset, while the variable **pool\_size** is the size of the resulting pool.

## Normalization

Before activation, normalization is applied to regulate unbounded activation to refrain output layers from over-expanding (Anwar, 2021). There are two approaches present in normalization. The first approach is the Local Response Normalization, which is a non-trainable layer that square-normalizes the pixel values in a feature map within a local neighborhood. The second approach is Batch Normalization, which will be the probable approach for the proposed model. This type of Normalization is a trainable approach to normalizing the data.

Before the hidden neuron return is fed to the activation function, they are processed in three steps. The first step is to normalize through nil mean and unit variance. Therefore, find the mean and variance by calculating based on the entire mini-batch output, then normalize the mini-batch by subtracting the mean and dividing it with the variance. The second step is to introduce two trainable parameters ( $\gamma$ : scale\_variable and  $\beta$ : shift\_variable) to scale and shift the normalized mini-batch output. The third step is to feed the scaled and shifted normalized mini-batch to the activation function.

## Activation

The activation function introduces non-linearity so CNN can efficiently map non-linear complex mapping between the input and the output. There are different types of activation functions. The first kind is non-parametric or static functions. The examples that fall under this category are Linear and ReLU.

$$RLi = \max(0, x_i) \quad (3.9)$$

The ReLU function used for the convolutional layer of the model has one variable, wherein the variable  $\mathbf{x}_i$  pertains to a single input. This function's role is to delete negative weights and replace them with zeroes instead.

The second set are the parametric functions. ELU, tanh, sigmoid and Leaky ReLU fall under the parametric functions and are defined as functions that use independent variables known as parameters.

$$SMi = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}} \quad (3.10)$$

The softmax activation used for the dense layer of the model has a single variable  $\mathbf{x}$  which signifies all inputs from the data. The function takes a single input  $\mathbf{x}_i$  and divides it by the summation of all inputs from the data.

### 3.4.2 Output Layer

Once the CNN model is defined, a loss function needs to be picked that quantifies how far off the CNN prediction is from the actual labels. This loss is then used in the gradient descent method to train the network variables. Similar to the activation functions, there are options available for the loss functions. The first category is called the Regression Loss Functions and the second category is called the Classification Loss Functions.

The Regression Loss Functions contains examples such as Mean Absolute Error, Mean Square Error, and Huber Loss wherein the estimated value and labels are real numbers. The Classification Loss Functions contains examples such as Cross-Entropy wherein the estimated value and labels are probability ranging from 0 to 1, and Hinge Loss wherein the estimated value and labels are real numbers.

### 3.5 Time Delay Neural Network

The Time Delay Neural Network, or TDNN in short, is a model that is capable of efficiently capturing temporal information. Since TDNN can capture temporal contexts, the TDNN can predict emotions despite its dynamic nature (Kumawat & Routray, 2021). Unlike the Recurrent Neural Network or Long Short-Term Memory, the TDNN can perform with less computational costs and faster parallelization ability during model training (Zhou & Beigi, 2020).

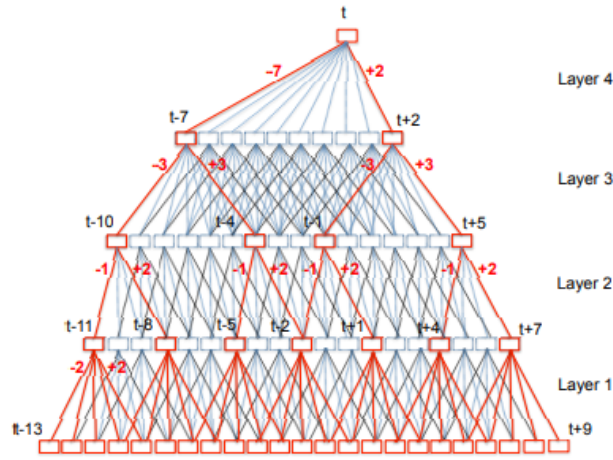


Figure 3.13: Computation of TDNN with sub-sampling (red) and without sub-sampling (blue + red) (Peddinti, Povey & Khudanpur, 2015)

TDNNs can also achieve less computational costs through sub-sampling at each layer, which is a process that involves input data and transforming it into a smaller form of data. This can be performed by using the kernel to stride over non-contiguous feature vectors and transforming each frame into smaller vectors with the use of neighboring data (Tchistiakova, 2019). Furthermore, in a TDNN that involves convolutions, the kernel weights are consistent and shared per layer and so the model is forced to learn invariant feature transforms (Peddinti, Povey, & Khudanpur, 2008).

In Figure 3.13, the bottom row of the architecture contains the input. The number of frames depends on the acoustic features of the input. Each frame in every layer is a column vector. The column vector represents a single time step in the audio signal, while the row vector is the feature values. The network slides over the signal and performs a convolution operation, which involves using a kernel or filter, that transforms the output (Tchistiakova, 2019).

The input vector can be also seen as  $x_t \in \mathbb{R}^m$ , where the vector contains real numbers. There will be a series of vectors, where each vector represents a one-step time,  $t$  of the speech. Therefore, the input features is seen as  $X \in \mathbb{R}^{m \times t}$ . The kernel or filter is seen as  $W \in \mathbb{R}^{m \times l}$ , where the dimensions of the kernel has the same height of  $m$ , and the width of  $l$ . The kernel moves  $s$  steps at a time, and the value of the stride,  $s$ , depends on how many steps the kernel is going to move at a time. The weights of the kernel are invariant, therefore the weights do not change.

$$o = \lfloor \frac{t - l + 2p}{s} \rfloor + 1 \quad (3.11)$$

In Equation 3.11, the value of the width depends on the number of times the kernel can fit at the length of the input sequence. There can be a padding of null values to the input of height  $m$  and length  $p$ . The padding helps in extending the area of which the kernel processes (Tchistiakova, 2019).

Present studies have used different variations of TDNN architectures. For example, a study made by Zhou and Beigi (2020) created a TDNN model with bottleneck layers using the entire IEMOCAP dataset. The performance of their model achieved an accuracy of 71.7%.

## Chapter 4

# Building a Speech Emotion Recognition Model

In the field of speech emotion recognition, there is a prevalence in the usage of posed datasets (Sun, 2020). In order to address this issue and assess the lack of naturalistic data in the context of speech emotion recognition, the researchers aim to create a model that is capable of detecting emotions in a naturalistic context.

The usage of end-to-end as the preferred framework and Convolutional Neural Network as the proposed algorithm will be documented in this section, as well as the general pipeline and layout of the models. The various model architectures will also be explained, as well as their final results and comparison.

### 4.1 Framework

In this section, the main machine learning pipeline of the models utilized in the study will be discussed. This includes the input of the models, as well as details concerning the splitting of the training and test sets, and the preprocessing procedures and the classifiers used for the models. Lastly the metrics outputted by the model will be discussed, as well as their significance.

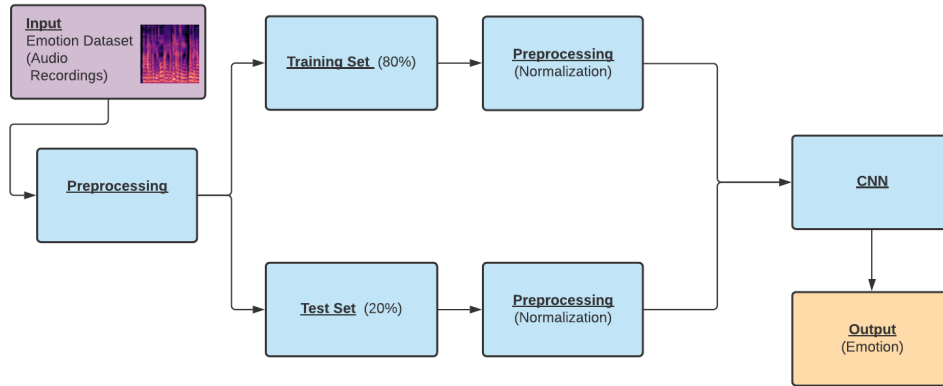


Figure 4.1: Machine Learning Framework for Speech Emotion Recognition

### 4.1.1 Input

During experimentation, a 40 MFCC feature set as well as 128 Mel-spectrogram features were extracted from the audio clips provided in the dataset. The feature sets were used as separate inputs for the models in order to assess which feature set would yield the best results for the models. Additionally, the feature sets were used as both a 1-dimensional input as well as a 2-Dimensional input, depending on the model used.

### 4.1.2 Training and Testing Sets

The models used during the experimentation used the train-test split available from the ScikitLearn Python library (Pedregosa et al., 2011). The data is split into 70-30, where 70% of the data goes to the training set while 30% of the data goes to the testing set. The models were trained using the training data before being validated using the testing data.

### 4.1.3 Classification

Six categorical emotion labels will be used for the models implemented for the study. These six categorical labels are based on the six basic emotions according to Ekman (Ekman & Friesen, 1978). However, it does not include contempt since IEMOCAP does not include it as an emotional label. Additionally, disgust is not

included because of the lack of ground-truth utterances labeled in the dataset (Busso et al., 2008). The labels used in the study, which are happy, sad, angry, neutral, surprised and fear, follow the categorical emotion approach. They are classified based on their unique behavior, trigger patterns and frequency distribution.

#### 4.1.4 Output

The output of the model consists of three metrics that shows the performance of the model. The three metrics are accuracy, loss and f1-loss. The metrics will be discussed individually.

##### Accuracy

Accuracy is a metric that is used to find out how many correct predictions the model made divided by the number of total predictions. However, the performance of the model should not be based on accuracy alone since the model can be accurate, but it can give a false sense that the model is performing well.

$$Accuracy = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \quad (4.1)$$

##### Loss

Loss is a metric that determines the difference of the expected outcome and the actual outcome that is produced by the model. The higher the value of the loss will signify that the model could not predict the outcome accurately, while a lower value of the loss will indicate that the model can predict the outcome correctly. There are different loss functions to use when calculating for the loss. It is important to consider the right loss function depending on the model and the data that is being fed to the model.

All of the models in this study use the Categorical Cross-Entropy as its loss function. The Categorical Cross-Entropy is used when the model needs to classify more than two labels. The function below is the formula to get the loss of a particular instance.



$$L = - \sum_{j=1}^M y_j \log(\hat{y}_j) \quad (4.2)$$

To calculate the loss, it is the negative summation of natural log of the prediction value provided by the softmax activation function.  $M$  refers to the number of classes or labels,  $y_j$  pertains to the expected outcome, and  $\hat{y}_j$  is the outcome produced by the model.

### **F1-loss**

$$precision = \frac{true\ positive}{true\ positive + false\ positive} \quad (4.3)$$

$$recall = \frac{true\ positive}{true\ positive + false\ negative} \quad (4.4)$$

F1-score is a metric that involves the true positive, true negative, false positive and false negative. F1 is based on two metrics, which are the precision and recall. Precision is calculated by true positive divided by the sum of true positive and false positive. Recall is calculated by true positive divided by the true positive and false negative.

$$f_{score} = \frac{2 * precision * recall}{precision + recall} \quad (4.5)$$

According to Maiza (2019), there are two ways to use the F1-score as an evaluation metric. The first option is to maximize the F1-score through threshold. The second option is to insert the F1-score into a loss function. However, the F1-score is not differentiable. Therefore, the F1-score in itself can't be used as a loss function. However, if the true positives, true negatives, false positives and false negatives are treated as a continuous sum of likelihood values by using probabilities, then it is possible.

$$f_{score} loss = 1 - f_{score} \quad (4.6)$$

In Equation 4.6, to simply calculate for the f1-loss, the f1-score should be subtracted from 1. Therefore, the range of values for the f1-loss is from 0 to 1.

The higher the value of the f1-loss indicates that the model is not performing well, while the lower the value of the f1-loss shows that the model is performing well.

## 4.2 Dataset Details

The interactive emotional dyadic motion capture database, or IEMOCAP, is an English dataset that contains both motion capture videos as well as utterances taken in order to capture the full image of human emotion. IEMOCAP includes both scripted and naturalistic utterances. The whole dataset contains over 12 hours worth of videos and recordings, with the naturalistic portion having a total of 4,784 utterances divided per turn level, described as the duration when an actor is speaking (Busso et al., 2008).

### 4.2.1 Dataset Contents and Environment

IEMOCAP is composed of 10 participants with a distribution of five male and five female participants. Participants were paired during data recording, which was conducted in an indoor lab environment to naturalize the emotions expressed during each session through the use of dyadic interaction. Additionally, 54 markers and two wristbands were attached to the participants in order to capture facial and gestural movements. Participants were asked to performed scripted emotional pieces as well as act out emotions originally from one of five emotion labels, with the labels being happiness, sadness, anger, frustration and neutral.

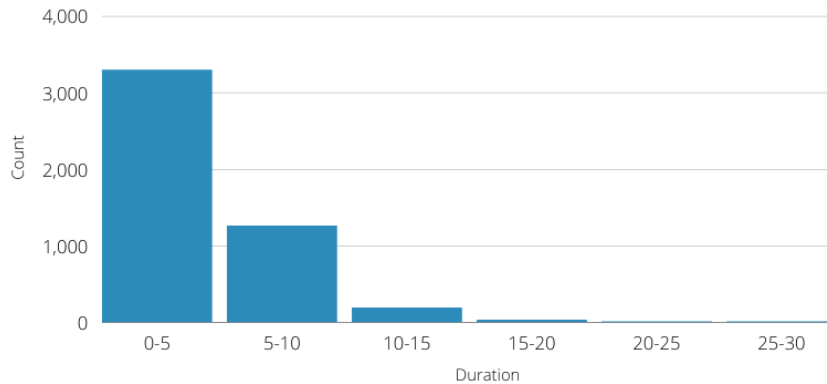


Figure 4.2: Histogram for Number of Utterances per length

A histogram was plotted highlighting the count of utterances based on duration. The results show that from the dataset, 3299 utterances have a duration of between 0-5 seconds while 1259 utterances are between 5-10 seconds long. There are 189 utterances with a length of 10-15 seconds, and 29 utterances with a duration of 15-20 seconds. Finally, there are 5 utterances that range from 20-25 seconds, and only 3 utterances that have a duration between 25-30 seconds. Based on the plotted histogram, it can be stated that the dataset is primarily comprised of utterances between 0 to 10 seconds long, with a large portion of the majority being made up of utterances with a length equal or less than 5 seconds. Utterances beyond 10 seconds in length only attribute to 4.72% of the dataset.

The researchers of IEMOCAP converted the videos into audio signals with a sampling rate of 16kHz. The the dataset averages 4.5 seconds per utterance, and the total duration of all utterances is over 12 hours. The dataset was annotated by six assessors, after which the dataset was also annotated by six of the 10 participants who provided the utterances present in the dataset. The utterances used from IEMOCAP were divided from each conversation so that each audio data only captures the speech segment in which there is only an instance of a single speaker on a turn basis. This means that there are no overlaps in the segments used for the experiments.

### 4.2.2 Emotion Labels and Distribution

IEMOCAP follows the discrete emotion theory, which states that there are only a set number of semantic primes, termed as ‘basic’ emotions, that are all relatively fixed regarding trigger patterns, how it is expressed as well as personal experiences to the emotion (Jeon, 2017). The emotion labels annontated in the dataset are *happy*, *sad*, *anger*, *frustration*, *excited*, *disgust*, *fear*, *surprise*, and *neutral* (Busso et al., 2008).

The division of utterances will include an equal amount of utterances per each emotion labels that were chosen for the study, with each label having 500 utterances. This distribution of labels was done to achieve as much proportionality among labels. It is also to avoid over-fitted results toward labels that are much greater in quantity over other fewer labels.

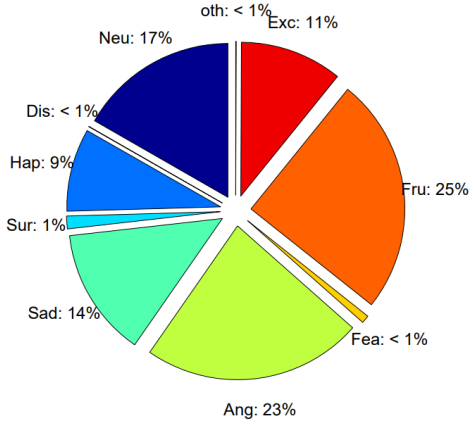


Figure 4.3: Naturalistic Emotion Label Distribution Busso et. al., 2008

### 4.2.3 Pre-processing and Feature Extraction

Each audio wav file was loaded using *pydub.AudioSegment.from\_file*, and then normalized using *pydub.effects.normalize* so that each audio signal is boosted up to 5 dBFs from the maximum volume (Robert, Webbie, et al., 2018). Each audio file was then trimmed using *librosa.effects.trim*, removing the silence at the beginning and end portions of the audio clip (McFee et al., 2015).

In order to perform length equalization for all data, we performed audio padding on each audio data. To achieve this, we iterated over the naturalistic audio clips to search for the audio clip with the longest duration. We found that the audio with the longest duration lasted for 29.1 seconds, and so we computed for the total frames of the audio clip by getting the total duration of the audio frames using *Wave.read.getnframes* and dividing it by the audio framerate which is obtained using *Wav.read.getframerate* (Van Rossum & Drake, 2009). The resulting computation showed that there is a total of 445952 audio frames on the audio data with the longest duration. We then padded the right side or the end of each audio file with silence, using 445952 audio frames as the maximum length for the padded audio data (Greenberg, 2022).

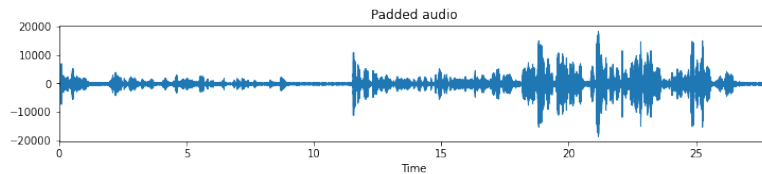


Figure 4.4: An Example of a padded audio clip

An example of a padded audio data is visualized in Figure 4.4. It is an example of an audio clip where audio padding was performed, wherein silence is padded on the right side or the end of an audio data to match the length of the longest trimmed audio file in the dataset. This process is also used on all the other audio utterances used in the study (Greenberg, 2022).

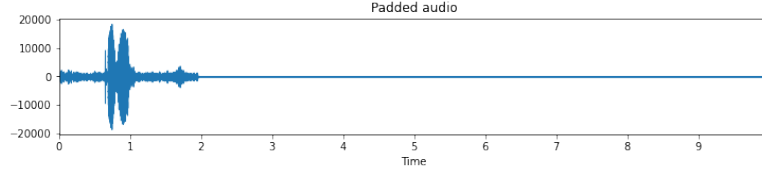


Figure 4.5: An Example of a padded audio clip for 10 seconds

Another set of data was also created using the same set of utterances and then padded to the length of 10 seconds. This is because in the naturalistic portion of IEMOCAP, utterances with a length less than 10 seconds composes of majority in the dataset. All utterances under the length of 10 seconds were taken, and right-side zero padding was performed on all of them in order to reach an equalized length of 10 seconds for the new set of data. An example of an audio data that was originally under 10 seconds, and then zero-padded with silence to reach 10 seconds is visualized in Figure 4.5

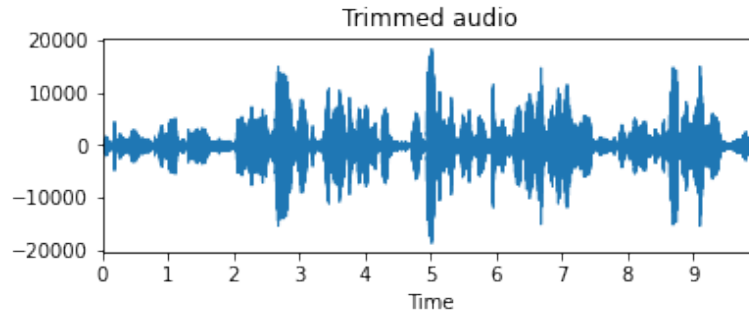


Figure 4.6: An Example of a clipped audio clip

For the clips that are longer than 10 seconds, we first find the specific frame in the audio where there the loudest signal was detected, indicating that the emotion in this particular time frame is most expressed. We then iterate over the audio data and get the surrounding signals until we reach a total of 160000 frames which is equal to 10 second audio at a sampling rate of 16KHz. There are also measures set so that when there is no longer an audio signal to the right, meaning that the iteration has reached the end of the clip, we instead iterate over the left side of

the signal for the remaining parts. The same rule applies when there is no longer an audio signal to the left, meaning that the iteration has reached the beginning of the clip. The same audio data that was previously padded in Figure 4.4 is used as an example of a clipped audio, which is visualized in Figure 4.6.

Two sets of features are then extracted from the audio data, using the function *librosa.feature.mfcc* to extract MFCC features and *librosa.feature.melspectrogram* for Mel-Spectrogram feature extraction (McFee et al., 2015). Both feature sets as well as the audio clips were then converted into binary files and loaded into a CSV file alongside the annotation files, which was done in order to sort the audio clips and feature sets into their respective annotated labels in the order it was stored in the folder.

The utterances under the Excited and Happy labels were merged together, as well as utterances labeled as Frustrated being merged with the utterances under the Sad emotion label, and due to the low count of *fear*, *surprise* and *anger* utterances, upsampling was performed on all three emotion labels, increasing their utterance count to 500 through the use of utterance duplication (Sahu, 2019). From the pre-processed dataset, 500 utterances from the *happy*, *sad*, *anger*, *fear*, *surprise*, and *neutral* emotion labels were taken, resulting in a dataset containing 3,000 utterances split evenly between 6 emotion labels being used for experimentation proceedings.

#### 4.2.4 Dataset Features

The resulting MFCC feature set extracted from each 29-second audio file has a shape of  $[40, 872]$ , with 40 referring to the number of Mel-frequency cepstrum coefficients and 872 referring to the number of features per coefficient. The resulting feature set for Mel-Spectrogram features has a shape of  $[128, 872]$ , with 128 referring to the number of Mel bins and 872 referring to the number of features for each Mel bin. The same applies for 10-second audio files, where the MFCC has a shape of  $[40, 313]$ , and the Mel-Spectrogram feature set has a shape of  $[128, 313]$ . Both feature sets were used for training and testing the model. Mel Frequency Cepstral Coefficients or MFCC features are the spectral density of an utterance based on a Fourier transform formula (Nair, 2018).

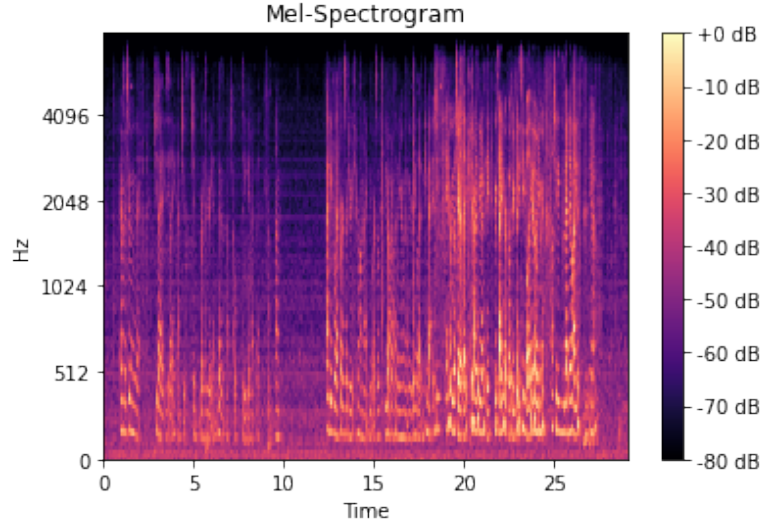


Figure 4.7: An Example of a padded audio clip

The Mel-Spectrogram features can be converted into decibel units (dB) using *librosa.power\_to\_db*, and the decibel values can be used to visualize a Mel-Spectrogram image using *librosa.display.specshow*. An example of a Mel-Spectrogram image that is produced using a sample Mel-Spectrogram feature set for our study is visualized in Figure 4.7. A Mel-Spectrogram image provides a visual representation of pure frequencies of a given signal concerning time (Chaudhary, 2020) and contains an x and y-axis, wherein the x-axis pertains to the Time and the y-axis corresponds to the range of frequencies in Hertz ( $Hz$ ), with the colors represent the magnitude of a frequency at a specific time.

#### 4.2.5 Libraries and Tools

Librosa is a Python package that is widely used for audio processing and analysis. It provides functionalities that helps in recovering information from given audio files (McFee et al., 2015). Librosa supports multiple audio codes, but **.wav** files are widely used for audio data analysis. One functionality of Librosa is displaying of mel-spectrogram figure through the use of *librosa.display.specshow* function.

Pydub is a Python library that is used with wav. files that allows these files to be edited and manipulated(J. Roberts, 2011). Pydub was used during preprocessing in order to normalize the data, by using the *pydub.effects.normalize* function.

The library *matplotlib* was utilized by using its function *matplotlib.pyplot.axis* and setting it to *off* to display the mel-spectrogram graph such that only the figure was displayed without the axis lines. The figure is then saved as *.jpeg* image file with fixed dimensions. These image files are then used as the training and testing data.

## Feature Extraction Tools

The feature extraction tools that were used to extract audio features from naturalistic audio from IEMOCAP is developed by Librosa. This section describes the specific functions that were used for feature extraction.

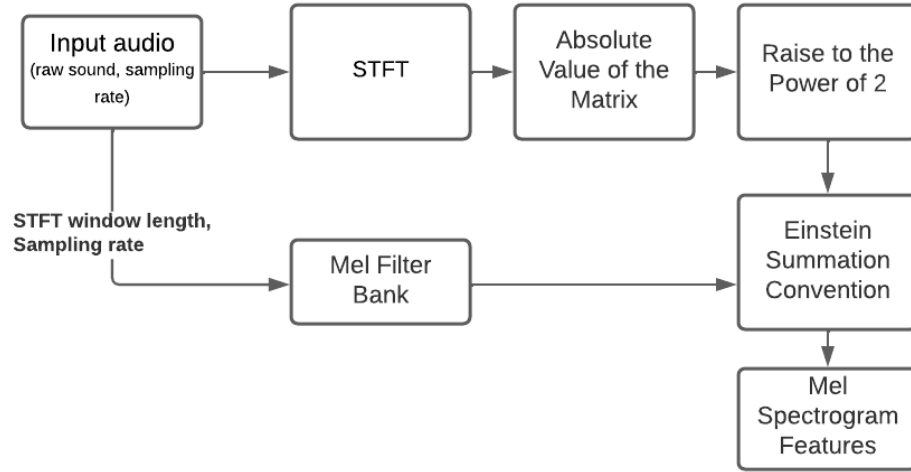


Figure 4.8: Mel-Spectrogram Feature Extraction Pipeline

The Librosa library was used to extract both the MFCC and Mel Spectrogram feature sets which are used as input. To illustrate this, for Mel-Spectrogram feature extraction or *librosa.feature.melspectrogram*, the function first takes in the audio time series as input, taking both the audio and sampling rate. The function takes one of the parameters which is the STFT window length, which has a default value of 2048, together with the sampling rate, which is 16KHz, into a Mel filter Matrix. The function then takes the raw audio and performs short-time fourier transform using Hann window function and a hop length of 512. The absolute value of the resulting Matrix from the STFT algorithm is then computed, and then the computed matrix is then raised to the power of 2 to get the squared magnitude of the STFT matrix. Finally, Einstein summation convention is performed on the



resulting Matrix as well as the Mel filter matrix using matrix multiplication, with the resulting matrix being the Mel Spectrogram features (McFee et al., 2015).

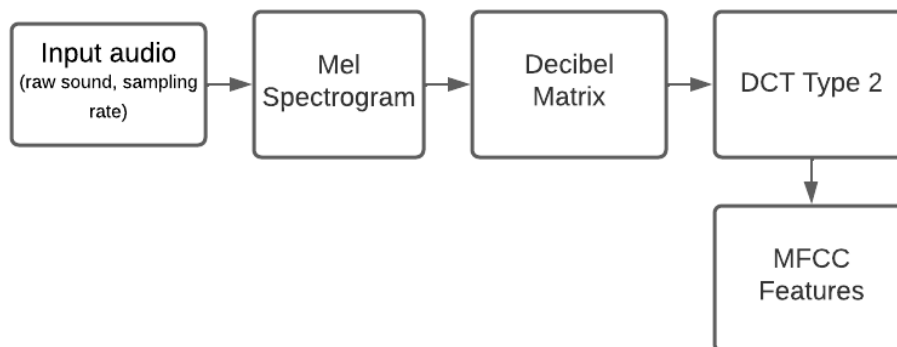


Figure 4.9: MFCC Feature Extraction Pipeline

For MFCC feature extraction or *librosa.feature.mfcc*, the process begins by extracting the Mel-Spectrogram features. After this, the function converts the mel-spectrogram into decibels before applying Discrete Cosine Transform Type 2 on the converted decibel matrix. The resulting transformed matrix is the MFCC feature-set (McFee et al., 2015). This illustrates that in Librosa, the MFCC features are primarily derived from the Mel-Spectrogram feature-set.

## 4.3 Traditional Machine Learning

Machine learning refers to an artificial intelligence implementation that allows systems to take a set of data and learn from it over time, independently improving its ability to perform a certain set of tasks or objectives as it continuously learns from the provided data, often split into data used to train the system and previously unknown data used to test it (Selig, 2022).

The study uses machine learning algorithms and models based on a study by Sahu in 2019, who used the entire IEMOCAP dataset on various ML algorithms to test their individual performances (Sahu, 2019). The models present in the study were individually reconfigured based on the parameters explicitly set in order to further optimize the models and increase model performance based on the model input. Multiple configurations based on certain parameters were tested using both the MFCC and Mel-Spectrogram input extracted from the 10-second utterance

database, with the resulting performance metrics being compared with each other to verify the best performing model. Additionally, all models used both MFCC and Mel-Spectrogram as input, extracted from the the pre-processed IEMOCAP naturalistic subdataset containing equalized 10-second utterances.

All machine learning models were reconfigured based on the existing parameters that were explicitly set by Sahu in his research in 2019 (Sahu, 2019), and as such still include room for improvement as the parameters that were set to default were not reconfigured or experimented on.

### 4.3.1 Random Forest Classifier

Random Forest Classifier is a supervised machine learning algorithm. It consists of individual decision trees that learn on various sub-samples provided by the dataset. It uses averaging to improve the accuracy and prevents the model from overfitting (Pedregosa et al., 2011). In this experiment, the Random Forest Classifier is implemented through the use of a Python library called Sklearn.

Two parameters in Random Forest Classifier were used based on a study made by Sahu (2019). *n\_estimators* and *min\_samples\_split* are experimented to find the best performance. *n\_estimators* determines the number of trees in the forest, and *min\_samples\_split* determines the minimum number of samples required to split an internal node. The rest of the parameters were set by default. The resulting values of the parameters were set at 800 *n\_estimators* and 4 *min\_samples\_split* for the MFCC feature input while Mel-spectrogram feature input were set to 400 *n\_estimators* and 4 *min\_samples\_split*.

### 4.3.2 XGBoost Classifier

XGBoost Classifier is a distributed gradient boosting library. It implements a machine learning algorithm using a Gradient Boosting framework. XGBoost provides a parallel tree through the use of an ensemble of weak prediction models in a form of decision trees (Pedregosa et al., 2011). The parameters used in this model were based from a study performed by Sahu (2019). The modified parameters are *max\_depth*, *learning\_rate*, *n\_estimators*, *subsample* and *booster*. The parameter, *max\_depth* determines the maximum depth of a tree. *learning\_rate* determines on how the weights are updated. *n\_estimators* determines the number of gradient booster trees. *subsample* determines how much data will the model use to grow the trees. Lastly, *booster* determines how the model behaves either through trees

or linear models.

For MFCC feature input, *max\_depth* is set to 7, *learning\_rate* is set to 0.008, *n\_estimators* is set to 1200, *subsample* is set to 0.8 and *booster* is set to gbtrees. For Mel-spectrogram feature input, *max\_depth* is set to 7, *learning\_rate* is set to 0.1, *n\_estimators* is set to 400, *subsample* is set to 0.4 and *booster* is set to gbtrees.

### 4.3.3 Multi-Layer Perceptron Classifier

A Multi-Layer Perceptron Classifier or MLP Classifier is one of the easiest to implement neural networks that are widely used in classification. Unlike most of the other classification algorithms used in the experiment, MLP classifier relies on an underlying neural network to perform the task of classification (Pedregosa et al., 2011).

In our experiment we implemented the MLP Classifier model with the use of Scikit-learn from the Python library, with the setup of hidden layer size set to 500, the Adam Optimizer as the optimizing algorithm, the *power\_t* set to 0.5 and *alpha* set to 0.0001, as well as the max iterations being set to 1000.

Additionally, to further optimize the model by Sahu, the *activation* or activation function and learning rate parameters were tweaked. The activation algorithm of a model is the function that adds non-linearity to a model and is the one that controls the output of the weights in the model (Sharma, 2021), while learning rate is split into two different parameters, the learning rate scheduler which is the pre-defined structure that changes the learning rate between epochs and the initial learning rate, which is simply the value of the model's initial learning rate (Li, 2021).

For experimentation, all activation functions available in the sklearn library being used. The learning rate scheduler was also tested using all available options in the sklearn library, while the initial learning rate was tested using the default value of 0.005 as the base with further experiments decrementing the learning rate by subtracting 2 from the from the greatest non-zero digit, with this being conducted in order to test the hypothesis of Brownlee, who stated that lower learning rate values allows the model to learn more slowly but also more accurately (Brownlee, 2019b).

For the resulting reconfigured parameters, when using MFCC as input, logistic regression was used as the activation function while the learning rate scheduler was set to constant and the initial learning rate was set to 0.0008. When using Mel-Spectrogram as input, the model instead uses Relu as its activation algorithm,

with the learning rate scheduler being adaptive and the initial learning rate being set to 0.001. All other parameters that were not mentioned were set the default.

#### 4.3.4 Support Vector Machine

A Support Vector Machine is a linear model for classification and regression problems. With many choices in hand, we imported and used Support vector classification from wherein the model has high capability to be utilized in multi-class classification on different datasets (Pedregosa et al., 2011).

The model is implemented with the use of Scikit-learn from the Python library, where the results are then one hot encoded in order to fit the `display_result` function created by us. For the model parameters,

The *kernel* parameter, which is the function used by the SVM model to solve and classify problems (Zoltan, 2022), was reconfigured due to the issues regarding the usage of a linear kernel to classify non-linear data. As such, all available kernels in the sklearn library were tested on both MFCC and Mel-Spectrogram, to assess which kernel would perform the best for each feature-set.

The resulting SVM model used for the study uses a Radial basis function when using MFCC as input, and a Radial basis function kernel when Mel-Spectrogram is used as input, with all other parameters set to default.

#### 4.3.5 Logistic Regression

A Linear Model library from Scikit-Learn was used to import a Logistic Regression classifier, an algorithm that involves modeling the probability of a discrete outcome given an input.

The Logistic Regression classifier implements regularized logistic regression in which regularization is applied by default (Pedregosa et al., 2011). The parameters include a maximum iteration set to 500, the multi-class is set to multinomial,

The solver parameter or optimization algorithm, which is the function that computes for the weight optimization (Brownlee, 2021), was reconfigured to assess which optimization algorithm would perform the best with the data. all optimization algorithms available in the sklearn library were used, and based on the tests conducted using different solvers, the solver for both MFCC and Mel-Spectrogram as input was set to the newton-cg optimization algorithm.

## 4.4 Model Architecture

The models present in the study were individually reconfigured based on the deep learning architecture that were explicitly set in order to further optimize the models and increase model performance based on the model input. Multiple configurations based on certain layers in the model architecture were tested using both the MFCC and Mel-Spectrogram input extracted from the 10-second utterance database, with the resulting performance metrics being compared with each other to verify the best performing model.

The deep learning models below are based on existing models and were further reconfigured to better read and classify the naturalistic data used for the experiments. Additionally, as the models used for the experiments are based on existing models with some configurations, it is noted that these models are not the optimal models for naturalistic speech emotion recognition. The experiments for these models were conducted such that each model was trained and tested until the training and testing accuracy, as well as the loss, matches. The model is only trained up to a certain epoch to prevent overfitting.

### 4.4.1 ResNet50

The ResNet50 model is an a variant of a Residual Network model containing a total of 50 layers. This model has been pre-trained using the ImageNet 2012 classification dataset that contains 1,000 classes and has been trained on the 1.28 million training images and evaluated on 50,000 validation images (He, Zhang, Ren, & Sun, 2015).

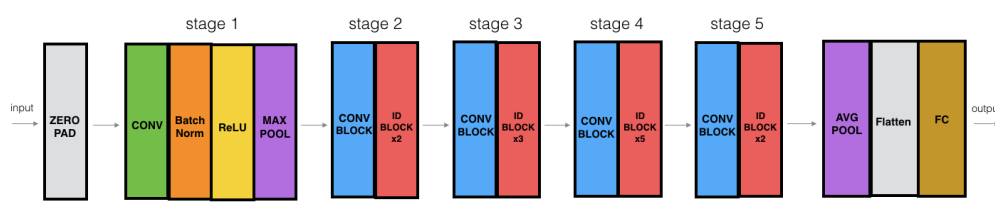


Figure 4.10: Resnet-50 Architecture (Manaswi, 2018)

For the model architecture, the input layer receives either the mel-spectrogram and MFCC featureset as 2D input for the ResNet50. The total 3,000-utterance subset of the dataset was split into training and testing sets. Then the ResNet50 model was imported with 48 convolution layers, 1 Max Pooling layer, and 1 Average Pool layer (Ji, Huang, He, & Sun, 2019). A dense layer was added for six

classifications of each emotion label with softmax as activation. The optimizing algorithm used for the model was the Adam Optimizer with a 0.001 learning rate and Categorical Cross Entropy as the loss function. The metrics used for the experiments is the categorical accuracy and f1-score. The model was trained with a batch size of 50, and was trained for both 50 epochs when using MFCC features and Mel-Spectrogram features.

#### 4.4.2 TDNN

##### TDNN with Dense Layers

The TDNN with dense layers is a modified model based from a study made by Kugler and Lehner (2019). In their TDNN model architecture, their model has an input, followed by a Dense layer, Batch Normalization layer, Dropout layer and a Dense layer. The Dense layer is responsible for receiving all of the information from the input layer. The Batch Normalization is responsible for stabilizing the learning process, normalizing the values and accelerates training. The Dropout layer helps in reducing the overfitting of the model and improve the generalization error. It goes through the dense layer again to acquire all of the previous nodes and it is responsible for classifying the labels based on what the model has learned. This TDNN architecture is based on the model of Kugler and Lehner (2019). It is implemented due to their construction of a TDNN with dense layers. However our own architecture is modified to be able to recognize naturalistic speech data. Experiments are conducted using this model to test and observe the performance of a TDNN model with dense layers instead of convolution layers.

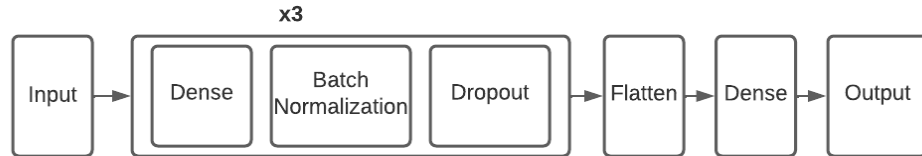


Figure 4.11: TDNN Dense Model Architecture

In this study, the Dense, Batch Normalization and Dropout layer has been run thrice sequentially to ensure that the model learns well. Hence, in 4.11, there is a big box surrounded by the three layers with a  $x3$  on top of it. The input for this model is both the MFCC and Mel-spectrogram features. The optimizing algorithm

is the Adam optimizer with a learning rate of 0.0005 to reduce the possibility of overfitting, while the loss function is the Categorical Cross Entropy. The Adam optimizer is an excellent optimization algorithm that handles sparse gradients on noisy problems. The metrics used for the experiments is the categorical accuracy and f1-score. The model was trained with a batch size of 50, and was trained for 35 epochs when using MFCC features, and 30 epochs when using Mel-Spectrogram features.

## TDNN with Convolution Layers

Another TDNN Model is used for the experiments. This model contains convolution blocks to learn the input features instead of dense layers. This model is used to test the performance of TDNN with convolution layers instead of dense layers. The model architecture consists of two convolution blocks. The first convolution block contains a convolution layer, an activation layer, max pooling layer, batch normalization and then dropout. The second convolution block only contains a convolution block, an activation layer, batch normalization and then dropout. The sequential model then contains global max pooling and then a dense layer with an activation function before producing the output.

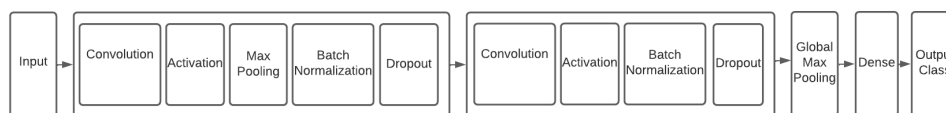


Figure 4.12: TDNN Convolution Model Architecture

To model follows different architecture depending on the type of input. For the experiments using MFCC features as input, in the first convolution block, the convolution layer has 32 filters and a kernel size of 6, the activation layer uses the activation function *relu*, the max pooling layer has a pooling size of 5, and then the dropout layer that follows the batch normalization has a dropout rate of 0.25. For the second convolution block, it follows the same values from the first convolution block, except that the convolution layer in this block has a filter of 64. After global max pooling, the dense layer that follows has 6 units of output with softmax as activation function.

For the experiments using Mel-Spectrogram features as input, in the first convolution block, the convolution layer has 64 filters and the same kernel size of 6, the activation layer uses the same activation function *relu*, the max pooling layer

has a pooling size of 5, and then the dropout layer that follows the batch normalization has a dropout rate of 0.25. For the second convolution block, it follows the same values from the first convolution block, except that the convolution layer in this block has a filter of 128. After global max pooling, the dense layer that follows has the same 6 units of output with softmax as activation function.

### 4.4.3 CNN + LSTM

The model is a modified model based from a study conducted by Zhao, Mao and Chen (2019). The model uses Convolutional Neural Network and Long Short-Term Memory (CNN LSTM) networks. The model that Zhao et al. performed uses CNN as the Local Feature Learning Blocks (LFLB) to learn the emotions based on speech found in the audio clips (Zhao, Mao, & Chen, 2019).

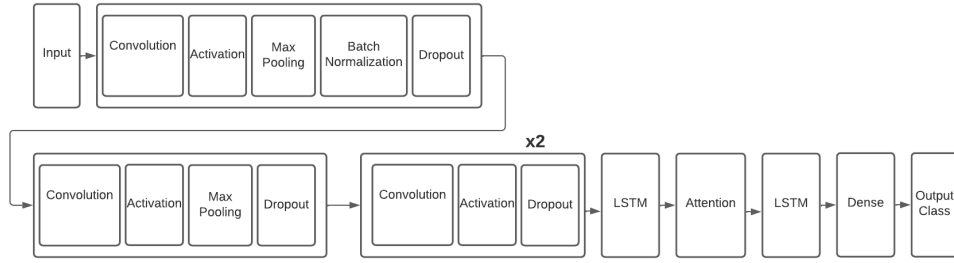


Figure 4.13: CNN + LSTM Model Architecture

In Zhao, Mao and Chen’s model, the LFLB consists of a Convolution layer, Batch Normalization layer, Activation layer and Max Pooling layer. Since there are four LFLB, the model runs the layers four times. The LFLB allows the convolution layer to perform the function of a learning kernel. The batch normalization layer normalizes the activations from the previous layer, and it helps with the stabilization of training. The activation layer, specifically using *relu* as its exponential linear unit, defines the output of the batch normalization layer. The max pooling layer divides the input into a set of non-overlapping regions and outputs the maximum value of each sub-regions. Once the four blocks have been run, it goes to the LSTM, which is specialized for processing a sequence of data values.

The model receives both the MFCC and Mel-spectrogram as its input. The input will go through the first and second LFLB where there is a convolution layer followed by activation layer, max pooling layer and a dropout layer. The dropout layer helps in preventing overfitting the model. In the first LFLB, Batch Normalization is performed before the dropout layer. After running through the



four Local Feature Learning Blocks, it will go through LSTM and then Dense layer before producing the output. The model uses Adam optimizer with a learning rate of 0.0005. The Adam optimizer is good optimizer for the experiments because it is an optimization algorithm that handles sparse gradients. The loss function is the Categorical Cross Entropy. The metrics used for the experiments is the categorical accuracy and f1-score. The model was trained with a batch size of 50, and was trained for 50 epochs when using MFCC features, and 60 epochs when using Mel-Spectrogram features.

#### 4.4.4 Base CNN

For the third set of experiments, a custom built convolutional neural network was built using the *keras* and *tensorflow* library available in Python. This model does not utilize any pre-existing models, and serves as a Base CNN model created by the researchers. Therefore, this model will be trained and tested with the naturalistic portion of the IEMOCAP dataset.

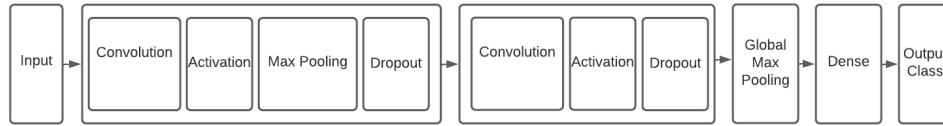


Figure 4.14: Base CNN Model Architecture for MFCC

The Base CNN architecture that is used for experiments using MFCC features consists of the first convolution block which contains a Convolution layer, an activation layer, a max pooling layer, and a dropout layer. It then follows another convolution block containing a convolution layer, an activation layer, and a dropout layer. The output of this convolution block then goes to a global max pooling layer and then a dense layer before it produces the output. The convolution filters for both convolution layers are increasing. The filter for the first convolution layer is 32, with a kernel size of 6. The activation layer that follows has an activation function of *relu* or Rectified Linear Unit. The max pooling windows has a pooling window size of 5. The dropout layer has a dropout rate of 0.01. The second convolution layer has a filter of 128, and a kernel size of 6. It has the same activation function for the next activation layer which is *relu*, and the dropout layer has the same dropout rate of 0.01. The dense layer that follows the global max pooling has 6 output units, corresponding to each emotion label, and it has an activation function of *softmax*.

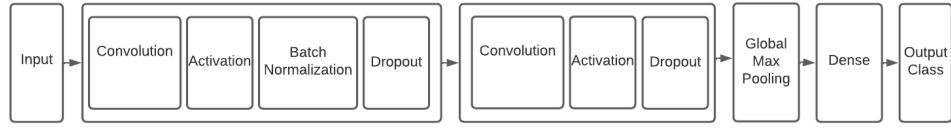


Figure 4.15: Base CNN Model Architecture for Mel-Spectrogram

The Base CNN architecture that is used for experiments using Mel-Spectrogram features follow the same architecture as the one used for the experiments with MFCC features. It also follows the same activation function *relu* for each convolution layer as well as the dense layer which uses *softmax*, the same dropout rate of 0.01 for the dropout layers, and kernel size of 6 for the convolution layers. The differences between this architecture with the one that was used for MFCC features is that the first convolution layer of this model has a filter of 64, and that this model uses batch normalization layer instead of a Max Pooling layer. This is because the batch normalization helps in stabilizing the model during training more than the max pooling. The reason for this is because the range of values of the Mel-Spectrogram features is very high.

The model receives both the MFCC and Mel-spectrogram as its input, in separate experiments. The model uses Adam optimizer with a learning rate of 0.0005. The loss function is the Categorical Cross Entropy. The metrics used for the experiments is categorical accuracy and f-score. The model was trained with a batch size of 50, and was trained for 100 epochs when using MFCC features, and 60 epochs when using Mel-Spectrogram features.

# Chapter 5

## Results and Analysis

This section of the paper presents the results and analysis of the conducted experiments using the naturalistic data from the IEMOCAP database. Multiple experiments were made on traditional machine learning models and on three separate neural network models. The traditional machine learning models are Random Forest Classifier, XGBoost Classifier, Multi-Layer Perceptron Classifier, Support Vector Machine and Logistic Regression. The neural network models are ResNet50, Convolutional Neural Network with Long Short-Term Memory and Time-Delay Neural Network. All of the experiments were done using the two different features as input. The two features are MFCC features and Mel-spectrogram features of 3,000 audio data. The number of audio data per emotion label is 500, with a total of six emotion labels used for the experiments. The emotion labels are Anger, Happiness, Sadness, Neutral, Fear, and Surprise. Two sets of pre-processed data were used during experimentation, one that was padded only to a maximum of 10 seconds, and one that was padded based on the length of the longest utterance in the dataset (29 seconds). Additionally, all models will be compared to a study by Sahu, who used the entire IEMOCAP dataset, using 5 different machine learning models to classify 6 emotions, namely Happiness, Sadness, Anger, Neutral, Fear, and Surprise.

The type of the input data for each model vary, in which ResNet50 requires 2-dimensional data, while CNN + LSTM, TDNN Convolution and TDNN Dense require 1-dimensional data. Because of this, the initial features that were extracted from 29-second audio data using *librosa.feature.mfcc* (for MFCC) and *librosa.feature.melspectrogram* (for Mel-Spectrogram) were in 2-dimensional form. Each data in the MFCC feature set is in the shape [40, 872] while each data in Mel-Spectrogram feature set are in the shape [128, 872]. This means that there is an exact 40 MFCCs and 128 Mel bins, and the same number of features (872)

for all audio data. This is because all audio data are trimmed and padded such that they would have the same length. The same also applies to the features extracted from 10-second audio data. The MFCC feature set from the 10-second data has the dimensions of [40, 313], and the Mel-Spectrogram feature set has the dimensions of [128, 313]. This 2-dimensional form of feature data of MFCC and Mel-Spectrogram was used as input for ResNet50.

All of the traditional machine learning models, CNN + LSTM, Base-CNN and TDNN require 1-dimensional data. Both the MFCCs and the Mel-spectrogram were transformed, so that the data can be used for the training and testing of the model. The MFCC and Mel-spectrogram feature sets were transformed by iterating over each data and computing for the corresponding mean of each feature using *numpy.mean* on each transposed data. Therefore, the resulting 1-dimensional data is then used as input as 40 MFCCs and 128 Mel bins for the aforementioned models.

For all experiments, the 3,000 feature data were split using the *sklearn.model\_selection.train\_test\_split* method from ScikitLearn and split into 70-30, with 70% of the data on the training set while 30% of the data is on the test set. For ResNet50, for each experiment with different feature set as input, the model ran using 50 epochs and 50 batch size, compiled with Adam optimizer with a learning rate of 0.001. For CNN + LSTM, Base-CNN, as well as TDNN, the models ran using batch size of 50, compiled with Adam optimizer with a learning rate of 0.0005. The model performance for all the models presented are measured using accuracy, loss, validation accuracy, and validation loss. Model performance per epoch is also visualized based on these metrics using matplotlib. The explanation of the calculation of the metrics can be found in Chapter 4.1.4.

Every model in the study is expected to learn from the input and predict the output accurately. The deep learning models have a graphical representation of the accuracy and loss metrics in respect to epochs. The accuracy graph is expected to start with a low accuracy. As the epoch progresses, the accuracy should start to increase until it reach its peak. As for loss, the value of loss is expected to be high in the beginning. As the epoch progresses, the loss will start to decrease. The visualization can be seen in Figure 5.1. The final results of accuracy and loss should be similar to the present studies, in order to show that the models can also learn data containing naturalistic emotions.

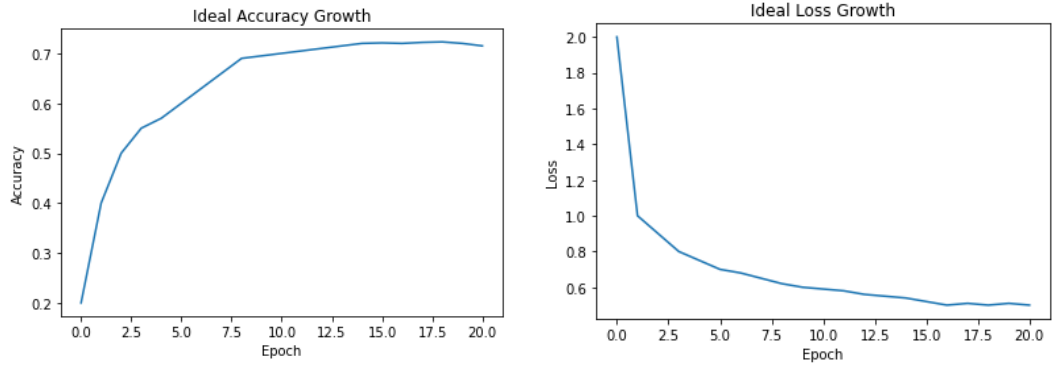


Figure 5.1: Visualization of Sample Ideal Graph of Accuracy and Loss

## 5.1 Random Forest Classifier

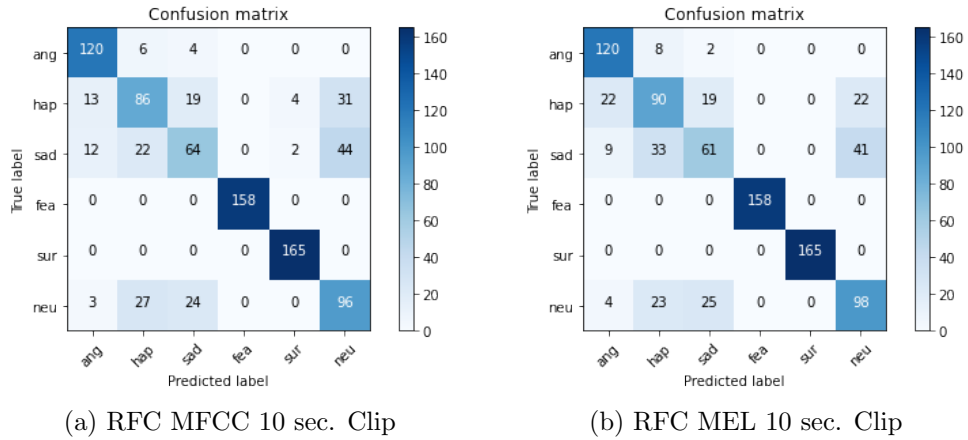


Figure 5.2: Random Forest Classifier Confusion Matrix

The RFC model using the MFCC features from the 10 second audio clip achieved an accuracy of 76.6%, an f1-score of 0.755, a precision score of 0.754 and a recall score of 0.762. Additionally, the RFC model using the Mel-spectrogram features from the 10 second audio clip achieved an accuracy of 77.2%, an f1-score of 0.76, a precision score of 0.759 and a recall score of 0.769.

The Random Forest Classifier was able to determine the emotion labels. Based from all of the confusion matrices presented in Figure 5.2, the RFC model is confused with happy, sad and neutral emotion labels. Fear and surprise are the two emotion labels that have the least amount of confusion.

In a study made by (Sahu, 2019), the Random Forest Classifier was experimented using the entire dataset of IEMOCAP, and the results are 56% accuracy, 0.56 f1-score, 0.572 precision and 0.573 recall. Comparing the results of Sahu’s study with this experiment, the experiment made in this study managed to have a better performance than Sahu. However, the most important detail to consider is that the model was still able to learn from the naturalistic features based from Sahu’s experiment using the induced dataset.

## 5.2 XGBoost Classifier

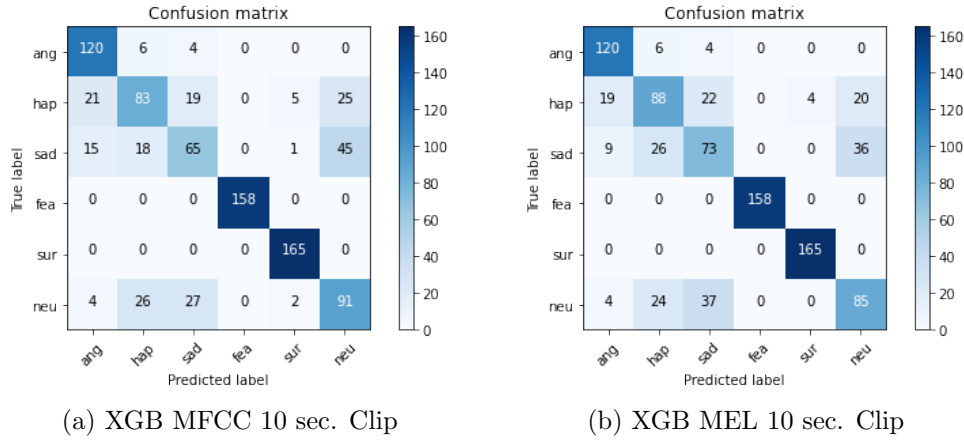


Figure 5.3: XGBoost Classifier Confusion Matrix

The XGBoost Classifier with MFCC features as the input performed an accuracy of 75.8%, an f1-score of 0.745, a precision score of 0.743 and a recall score of 0.754. Also, running the model using Mel-spectrogram features resulted in an accuracy of 76.6%, an f1-score of 0.756, a precision score of 0.753 and recall score of 0.762.

Based on the performance of XGBoost Classifier, the model is confused with happy, sad and neutral emotion labels. This is a similar behavior to the Random Forest Classifier. Based on a study made by (Sahu, 2019), the XGBoost results in Sahu’s model achieved 55.6% accuracy, 0.56 f1-score, 0.569 precision and 0.568 recall. The XGBoost Classifier model in this study performed much better, but the more important factor is that the XGBoost model was able to learn based on the naturalistic data input.

### 5.3 Multi-Layer Perceptron Classifier

The MLP Classifier using MFCC features as input performed with a test accuracy of 76.1%, an f1-score of 0.751, a precision score of 0.752, and a recall score of 0.756. When Mel-Spectrogram features are used as input, the model achieved a test accuracy of 70.6%, an f1-score of 0.679, precision score of 0.689, and a recall score of 0.699.

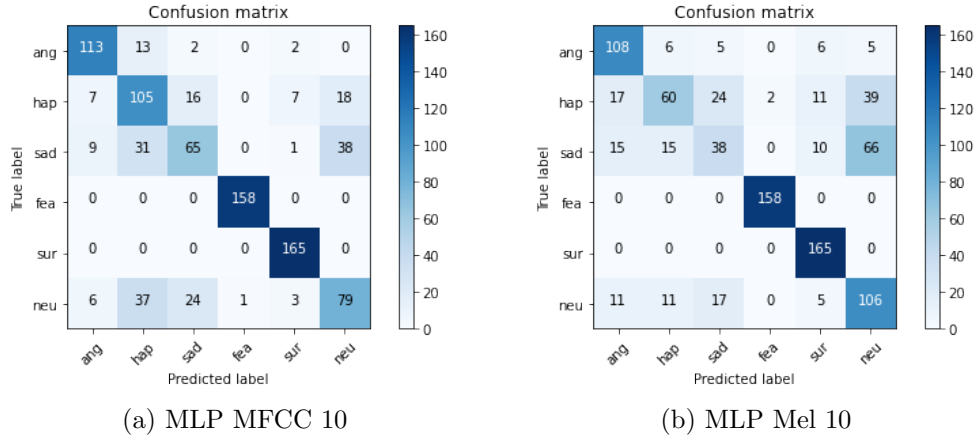


Figure 5.4: MLP Confusion Matrices

Based on the model performance of the MLP classifier using MFCC features, as well as its performance using Mel-Spectrogram features as its input, the model was able to properly classify the categorical emotions of the data. Compared to the model performance using MFCC features as input, the result of the MLP using MFCC features was higher in terms of accuracy, f1-score, precision and recall. It is also observed that the model is confused with the emotion labels of happy, sad, and neutral.

In an experiment from (Sahu, 2019), the MLP results in their model performed with 41.0% accuracy, an f1-score of 0.365, a precision of 0.422 and a recall of 0.359. Based on the results of the MLP Classifier model in this study, using MFCC features, as well as with Mel-Spectrogram features as input, the model in our study performed better than the model used in the study from Sahu, which was further improved with the reconfiguration of the activation and optimization algorithm as well as the learning rate.

## 5.4 Support Vector Machine

Using MFCC as its input, the model performed with an accuracy of 28.9%. The f1-score of the model achieved a score of 0.277, while the precision returned a score of 0.223 and recall of 0.291, indicating poor model accuracy. When using Mel-spectrogram as its input, the model performed with an accuracy of 57.1%. The f1-score of the model achieved a score of 0.566, while the precision returned a score of 0.577 and recall of 0.565, attributing the model's improvement through the use of the polynomial kernel to better fit the non-linear data being used .

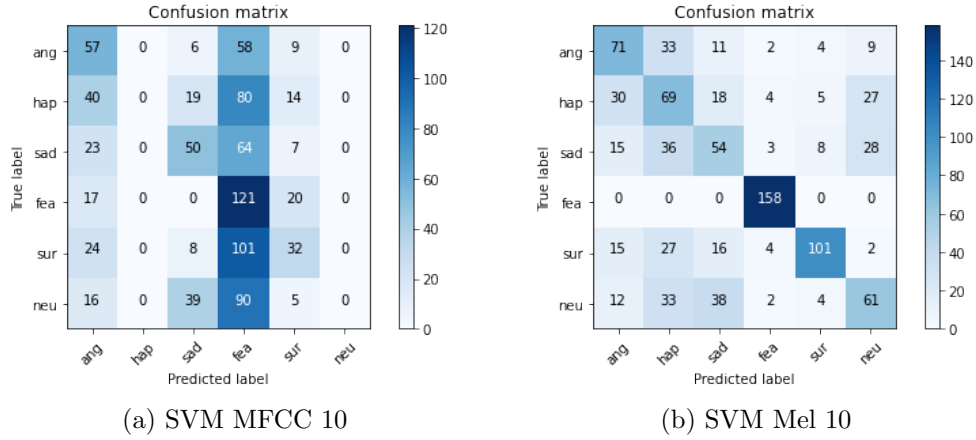


Figure 5.5: SVM Confusion Matrices

Based on the performance of the mode, only the SVM model with Mel-Spectrogram model is able to predict the emotion labels to a certain accuracy. However, in comparison to the other traditional machine learning models, the Support Vector Machine is vastly inferior to other models, with the SVM model using Mel-Spectrogram as input only being comparable to the results given by the Logistic Regression model.

The Support Vector Machine used from (Sahu, 2019) performed with 33.7% accuracy, an f1-score of 0.152, a precision of 0.174 and a recall score of 0.215. When comparing the results of the SVM model used in this study using the Mel-Spectrogram features, as well as the results with MFCC features as input, our SVM model performed better in terms of accuracy, f1-score, precision and recall score when using Mel-Spectrogram as input, but performed worse when using MFCC. This shows that SVM when using 1D Mel-Spectrogram as input, is able to classify categorical emotions when using naturalistic data from IEMOCAP to a degree.



## 5.5 Logistic Regression

For logistic regression, when using MFCC features as input, the model achieved an accuracy of 55.8% and an f1-score of 0.545. The precision score is 0.539 and the recall score is 0.556. When using Mel-spectrogram features as input, the LR model was able to achieve an accuracy of 60.2% and an f1-score of 0.584. The precision score is 0.579 and the recall score is 0.594.

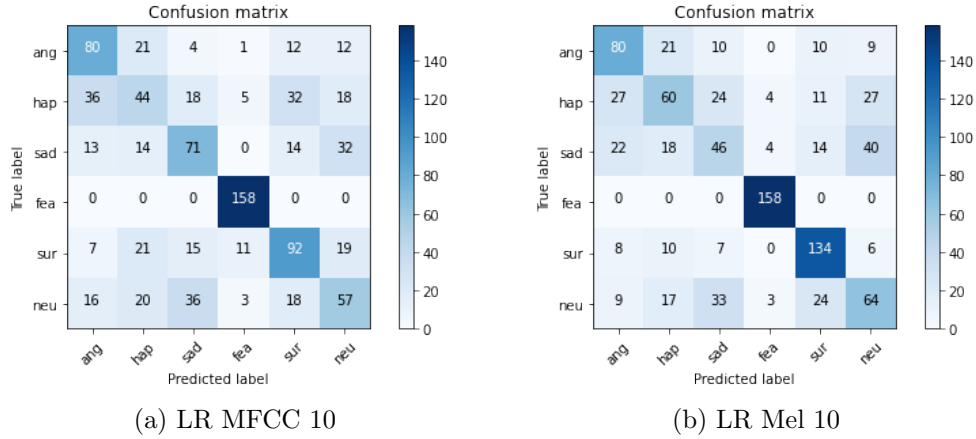


Figure 5.6: LR Confusion Matrices

Based on the model performance of Logistic Regression using both MFCC features and Mel-Spectrogram features, the model performed adequately in recognizing categorical emotions in a naturalistic sense, but are outperformed by the other machine learning algorithms used during experimentation with the exception of the Support Vector Machine Model, which achieved an accuracy score of above 70%.

When comparing to the logistic regression model by (Sahu, 2019), the model by Sahu with an accuracy rate of 33.4% is largely outperformed by the model used in the study with both MFCC and Mel-spectrogram is used as input, having an accuracy score of 55.8% and 60.2% respectively.

## 5.6 Analysis of Machine Learning Models

Overall, the performance of Machine Learning algorithms used in this study, which includes Random Forest Classifier, XGBoost Classifier, Multi-Layer Perceptron,

Support Vector Machine, and Logistic Regression performed with higher accuracy than the ML algorithms used in the study from (Sahu, 2019). A consistent pattern can be observed using the confusion matrices used to visualize the model performance of ML models, and that is the models almost always correctly classify data labeled as *fear* and *surprise*. A reason for this may be because of the up-sampling method used for the data in this study, in which audio data with 'fear', as well as audio data with 'surprise' labels were upsampled through utterance duplication which was also used by (Sahu, 2019), and that the upsampled data that were added to the dataset were too similar to each other.

It was observed that the Random Forest Classifier had the best performance out of all machine learning models. This is due to the RFC algorithm generating many decision trees rather than just a single one, which helps in making more accurate class predictions. RFC also is more appropriate with classification problems, which the prediction of emotion labels is regarded as, as well as being effective regardless of the size of the dataset (Yiu, 2021).

Additionally, all machine learning models were also tested using both the dataset containing 29-seconds and another dataset that only equalized the utterances to 10 seconds to reduce zero-padding. These experiments were conducted to know whether the reduction of zero padding will result with better model performance or not, with the models using both the MFCC and Mel-Spectrogram features from the datasets used. The results can be seen at table 5.1 and 5.2 below:

Name of Model	Accuracy (10s)	F1-score (10s)	Accuracy (29s)	F1-score (29s)
Random Forest Classifier	0.764	0.752	0.751	0.741
XGBoost Classifier	0.758	0.745	0.752	0.741
Multi-Layer Perceptron Classifier	0.761	0.751	0.68	0.65
Support Vector Machine	0.289	0.227	0.223	0.291
Logistic Regression	0.558	0.545	0.551	0.537

Table 5.1: Machine Learning Experiments 10s Vs 29s MFCC

From the results of both 10s and 29s datasets using MFCC, it can be said that the reduction of zero padding resulted in little to no difference when tested using MFCC as input. All models show similar results with almost no difference when tested using the two datasets with the exception of the MLP machine learning algorithm, which had a difference of roughly 7-8%.

Name of Model	Accuracy (10s)	F1-score (10s)	Accuracy (29s)	F1-score (29s)
Random Forest Classifier	0.77	0.758	0.767	0.754
XGBoost Classifier	0.762	0.751	0.766	0.755
Multi-Layer Perceptron Classifier	0.706	0.694	0.694	0.676
Support Vector Machine	0.571	0.566	0.577	0.572
Logistic Regression	0.602	0.584	0.600	0.582

Table 5.2: Machine Learning Experiments 10s Vs 29s Mel Spectrogram

For Mel-Spectrogram, the results show a similar pattern when compared to those using MFCC as input, showing little to no difference between all models. The performance metrics of the models using 10 second and 29 second data are similar to each other, with no significant difference between results.

## 5.7 ResNet50 NSER

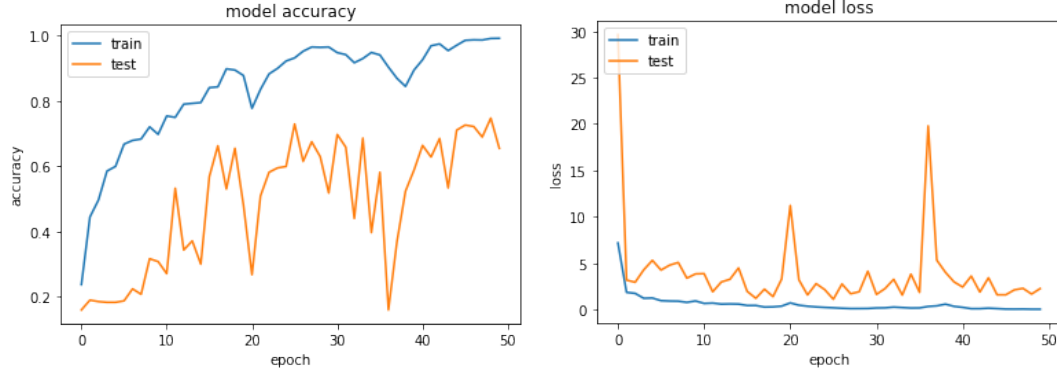


Figure 5.7: ResNet50 NSER MFCC Performance Results

The ResNet50 NSER model when using MFCC as input performed with a training accuracy of 99.18% and a validation accuracy result of 65.44%, a training loss of 0.0242 and a validation loss of 2.2547, and a training f1-score of 0.9866 and a validation f1-score of 0.6460.

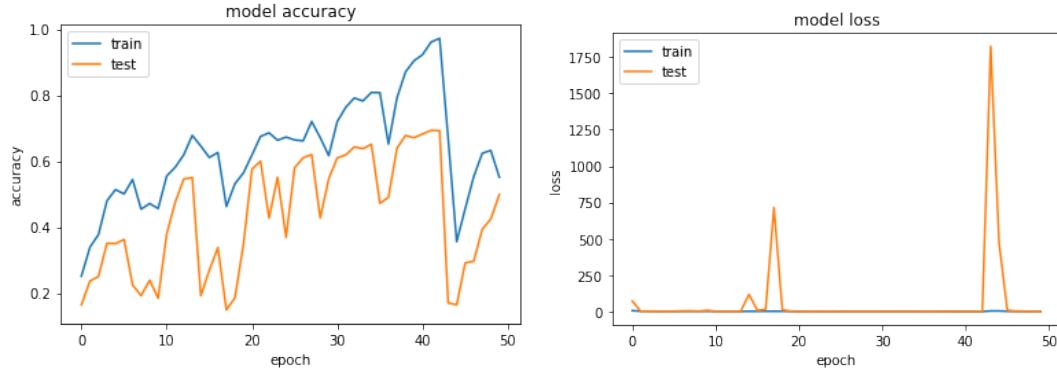


Figure 5.8: ResNet50 NSER MFCC Performance Results

As for the Mel-Spectrogram feature, the ResNet50 NSER model achieved a training accuracy of 56.29% and a validation accuracy of 49.89%. In terms of loss, the model performed with a training loss of 1.4822 and a validation loss of 1.4104. Lastly, in terms of f1-score, the model showed a training f1-score of 0.4788 and a validation f1-score of 0.3862. Both models ran for 50 epochs and a batch size of 50. Based from the results, the model learned best from the MFCC features.

Looking at the related literature, there are studies in the past that have used the ResNet50 model to test both large scale and posed datasets. a study made

by Ayadi and Lachiri in 2022 used a pre-trained ResNet50 model during their experiments with both song and speech. The model includes 50 layers split between 6 blocks, with the first block being a convolution layer with a kernel size of 49. This is then followed by 4 blocks, each block containing 9 , 12, 18 , and 9 convolution layers respectively, with the final block containing a fully connected layer. The model for their study uses RAVDESS, a posed dataset that includes both song and speech features, and was trained and tested to classify 6 emotion labels, namely Angry, Calm, Fear, Happy, Neutral, and Sad. The model performed with an accuracy of 55.52% (Ayadi & Lachiri, 2022b), achieving a lower accuracy score compared to the validation accuracy score of 65.44% by ResNet50 NSER when MFCC is used as input.

Another study made by Fan et al. in 2021 used a pre-trained ResNet50 model with the same architecture discussed previously, but this time placed an emphasis on the usage of large scale data. The dataset used was LSSSED, which is a recent naturalistic dataset that includes a large quantity of data compared to other datasets. The model was trained and tested to classify 4 emotion labels, namely Angry, Happy, Sad, and Neutral, and performed with an unweighted accuracy of 37.7% (Fan, Xu, Xing, Chen, & Huang, 2021), which was lower compared to the 65.44% and 49.89% validation accuracy scores of ResNet50 NSER when using either MFCC or Mel-Spectrogram as input.

## 5.8 TDNN NSER

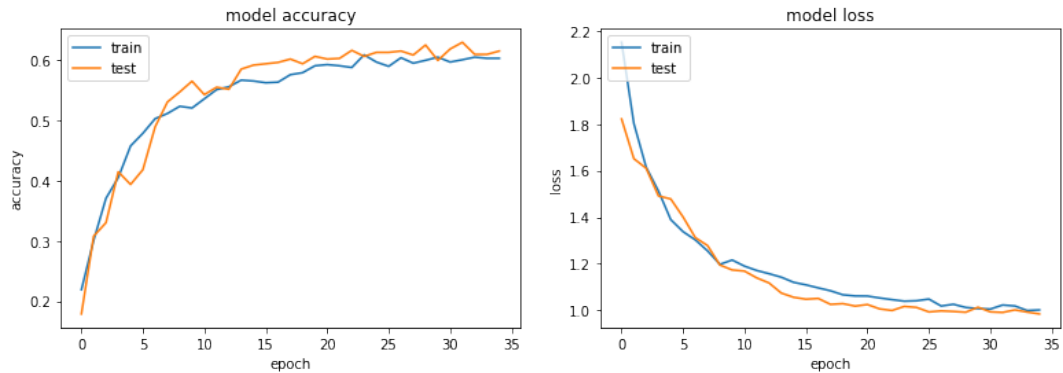


Figure 5.9: TDNN Dense NSER MFCC Performance Results

The TDNN NSER model that uses dense layers achieved a training accuracy of 59.61%, a validation accuracy of 61.44%, a training loss of 1.0142, a validation

loss of 0.9845, a training f1-loss of 0.4764 and a validation f1-loss of 0.4776 when using MFCC as its input.

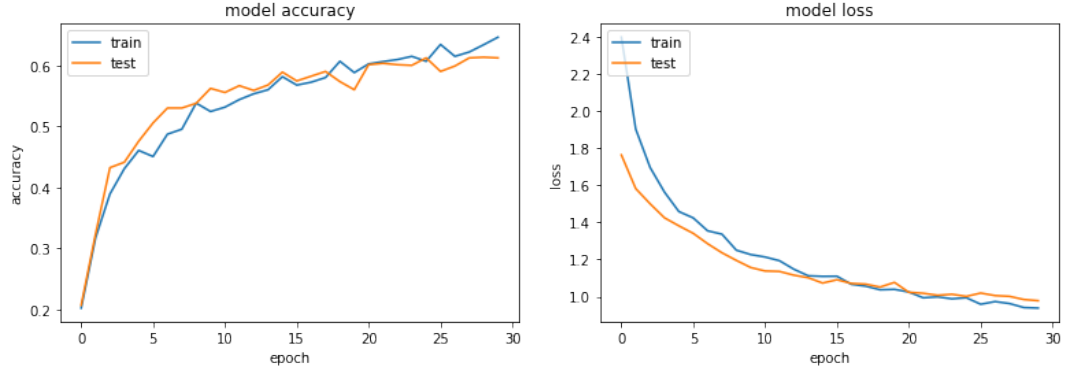


Figure 5.10: TDNN Dense NSER Mel-Spectrogram Performance Results

Additionally, the model that uses Mel-spectrogram as its input performed with a training accuracy of 64.59%, a validation accuracy of 61.22%, a training loss of 0.9209, a validation loss of 0.9780, a training f1-score of 0.5147 and a validation f1-score of 4734.

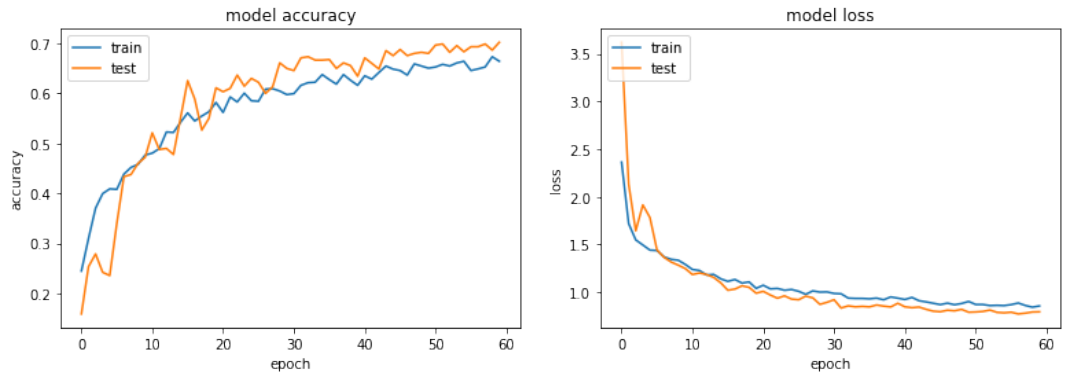


Figure 5.11: TDNN Convolution NSER MFCC Performance Results

When reconfigured to use convolutional layers instead of dense layers, the TDNN model was able to achieve a training accuracy of 66.83%, a validation accuracy of 70.22%, a training loss of 0.8652, a validation loss of 0.7951, a training f1-score of 0.5455 and a validation f1-score of 0.5387. with MFCC as input.

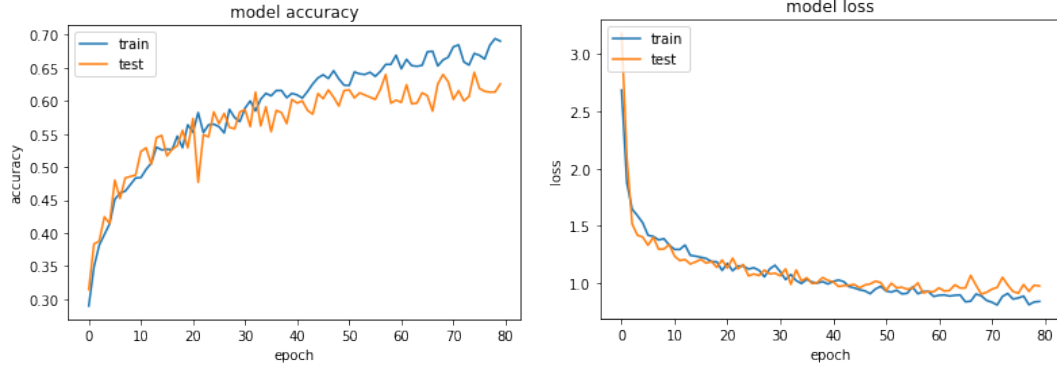


Figure 5.12: TDNN Convolution NSER Mel-Spectrogram Performance Results

When Mel-Spectrogram is used as input, the mode achieved a training accuracy of 69.80%, a validation accuracy of 62.56%, a training loss of 0.8088, a validation loss of 0.9737, a training f1-score of 0.5859 and a validation f1-score of 0.5239.

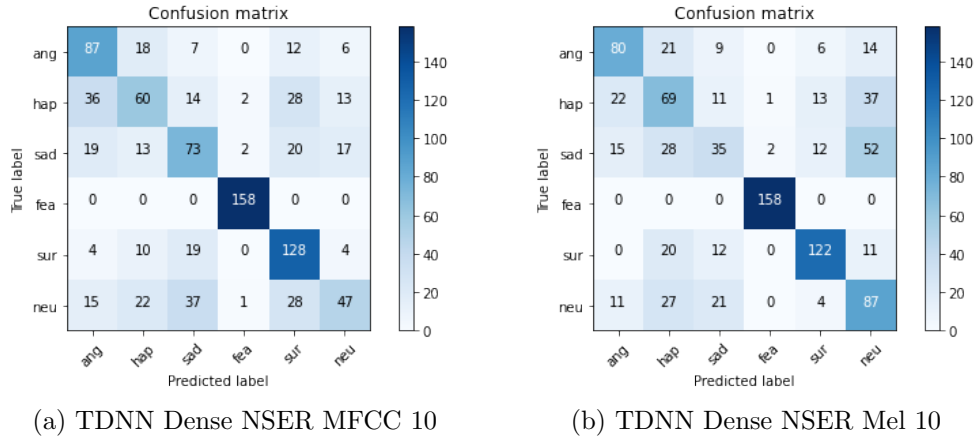


Figure 5.13: TDNN Dense NSER Confusion Matrices

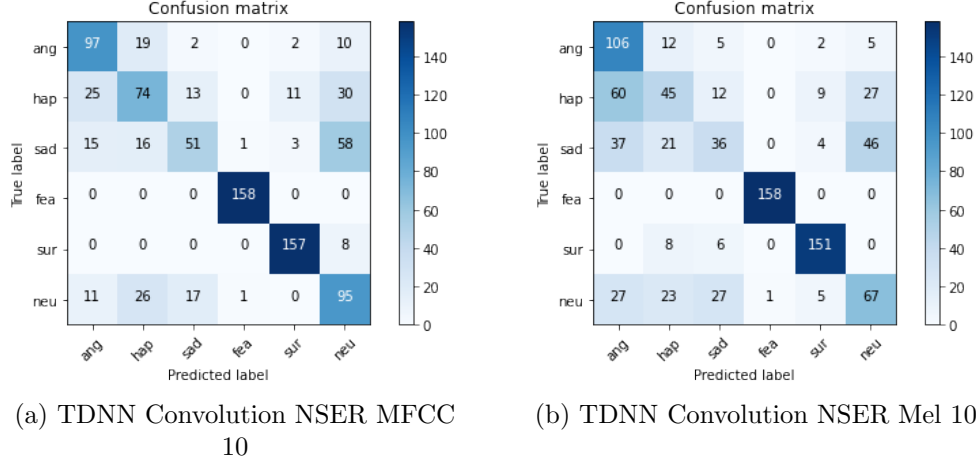


Figure 5.14: TDNN Convolution NSER Confusion Matrices

The TDNN NSER model using dense layers ran for 35 epochs when using MFCC and 30 epochs when Mel-spectrogram was used as input, with both variations having a batch size of 50, while the TDNN NSER Model using convolutional layers ran for 60 epochs when using MFCC, and 80 epochs when using Mel-Spectrogram, with a batch size also equal to 50 for both feature sets. Based from the results, the TDNN NSER model using convolutional layers achieved higher performance when using MFCC as input. In addition, the performance of the TDNN convolution NSER model using either feature set as input outperformed the TDNN Dense NSER model using both MFCC and Mel-spectrogram as input. It has also been observed that both models have trouble classifying both sad and happy utterances, as the confusion matrices for both matrices show a number of happy and sad utterances being misclassified as neutral, interestingly, while also showing signs of misclassification of happy and sad utterances, the TDNN Dense NSER model that uses MFCC as input also experiences neutral utterances being misclassified as either happy or sad utterances, with fewer correctly classified neutral utterances than the other TDNN NSER model variations. All models were able to accurately classify utterances labeled as fear or surprise.

From the related literature, there are studies that uses the entire IEMOCAP datasets to train and test their TDNN models. This study uses the improvised data found in the IEMOCAP dataset, which contains the naturalistic emotions. Comparing the present studies to this study, the models in this study performed with higher accuracy.

A study made by Kumawat and Routray (2021) uses the IEMOCAP data on two different TDNN architectures, which are the ECAPA-TDNN and X-Vector TDNN architectures. the ECAPA-TDNN model uses ECAPA, which uses 2 Con-



volution Blocks with 3 1-dimensional Squeeze-Excitation Residual Blocks, which are used to revise certain channel-wise features by modeling links between channels, in between. The output generated by the 3 SE-Res2Blocks are concatenated and fed to the lower convolution block, wherein stat pooling and batch normalization is applied to it. the ECAPA-TDNN architecture uses Adam as its optimizer and Additive Angular Margin (AAM)-softmax as its loss function. The X-vector architecture on the other hand uses a 1-D SE-Res2Block between convolutional layers similar to the ECAPA-TDNN, but only performs statistical pooling on the outputs from the TDNN before. Both models use MFCC as their input and uses the IEMOCAP dataset and 4 emotion labels, namely Angry, Happy, Sad, and Neutral. Results show that the ECAPA-TDNN and X-vector model achieved a validation accuracy of 50.71% and 58.67% respectively (Kumawat & Routray, 2021). Both TDNN Dense NSER and TDNN Convolution NSER models achieved higher accuracy scores than the literature mentioned.

A study conducted by Sarma et al. in 2018 used TDNN with statistics pooling in order to identify emotions through the use of raw signals. The architecture of the model includes a set of TDNN layers as well as a statistics pooling layer, outputting a set of means and standard deviations based on the input, which are then connected together and sent through a feed forward layer and then a softmax layer. The model uses IEMOCAP as its dataset, extracting a 23 MFCC feature set and using it to classify 4 emotion labels, namely Angry, Happy, Sad, and Neutral, with a resulting validation accuracy of 55.3% (Sarma et al., 2018), which was lower than the accuracy scores achieved by both the TDNN Dense NSER and TDNN Convolution NSER models.

However, a study conducted by Zhou and Beigi (2020) that uses ASR transfer learning and TDNN, with an architecture consisting of 13 TDNN layers. The Stride for each TDNN layer was explicitly set to 0 for layers 1 and 5, set to 1 for layers 2 and 4, and set to 3 for the layers 6 up to 13. The model also contains an ASR pre-training layer, where ASR pre-training was conducted alongside the final 12th and 13th TDNN layer for emotion detection. For the dataset used, the entire IEMOCAP dataset was used to train and test the model using MFCC with iVector-based features classifying 4 emotion labels, namely Angry, Excited, Sad, and Neutral. The model reached an accuracy of 71.7% (Zhou & Beigi, 2020), which is higher compared to the performance of the study's TDNN Dense NSER and TDNN Convolution NSER model, indicating the effectiveness of the usage of ASR pre-training on their TDNN model. It is important to note however that the model used in this study only used the naturalistic portion of the IEMOCAP dataset as well as classifying 6 emotions, which can indicate a lower performance compared to Zhou and Beigi's model that used the entire IEMOCAP dataset classifying only 4 emotion labels.

## 5.9 CNN + LSTM

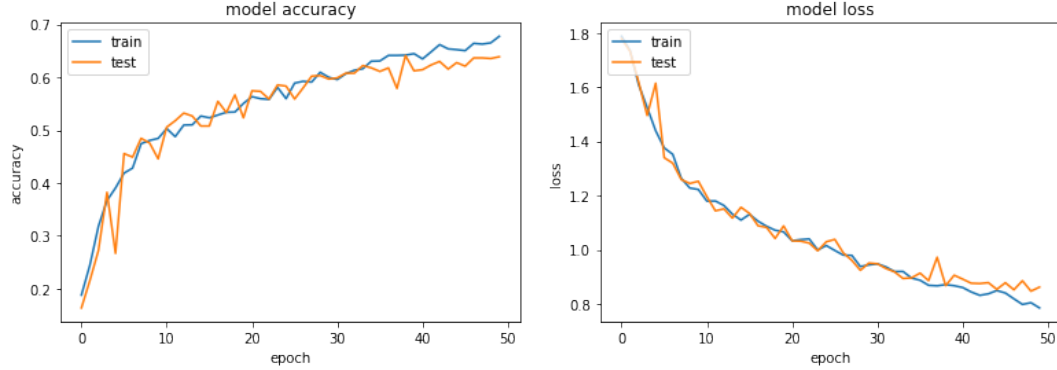


Figure 5.15: CNN+LSTM NSER MFCC Performance Results

The CNN+LSTM NSER model with MFCC as its input performed a training accuracy of 67.70%, a validation accuracy of 63.89%, a training loss of 0.7833, a validation loss of 0.8617, a training f1-score of 0.5686, and a validation f1-score of 0.5388.

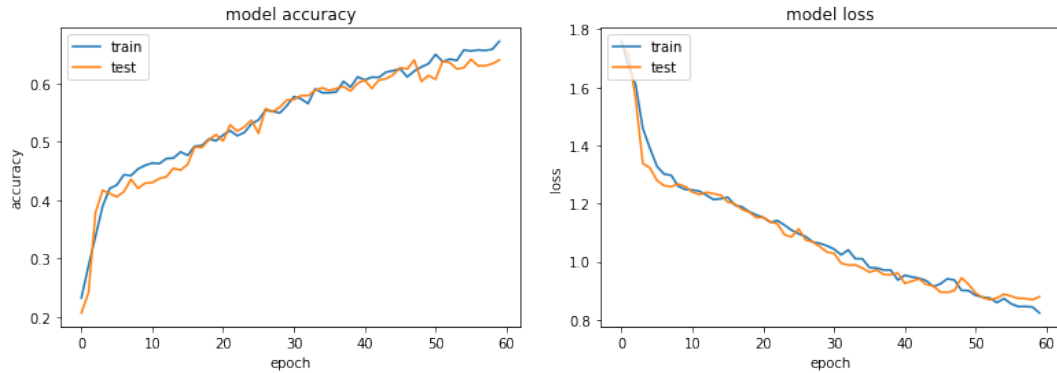


Figure 5.16: CNN+LSTM NSER Mel-Spectrogram Performance Results

When using Mel-spectrogram as input, the CNN+LSTM model achieved a training accuracy of 67.76%, a validation accuracy of 64%, a training loss of 0.8232, a validation loss of 0.8791, a training f1-score of 0.5335 and a validation f1-score of 0.5317.

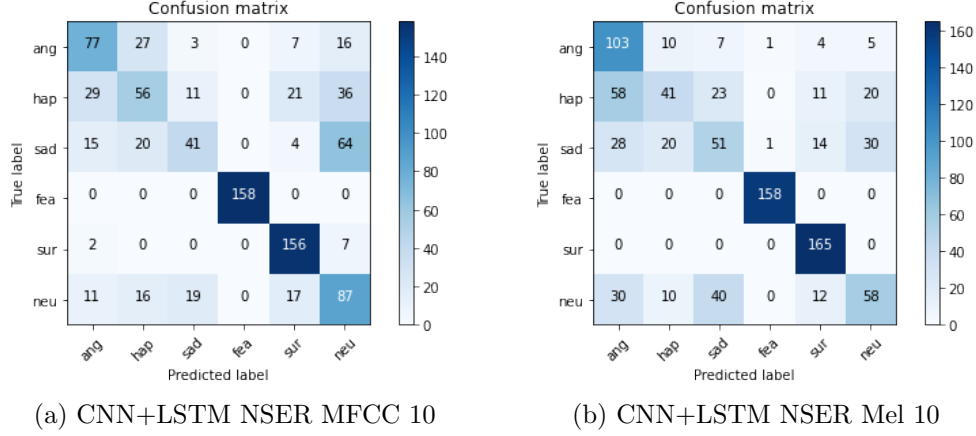


Figure 5.17: CNN+LSTM NSER Confusion Matrices

The model ran for 50 epochs when using MFCC as input and 60 epochs when Mel-spectrogram was used, with a batch size of 50 for both inputs. Based from the results, the CNN+LSTM NSER model learned well when using both MFCC and Mel-spectrogram feature sets, but performed slightly better when using Mel-spectrogram as input. Both variations of the model had issues classifying happy, sad and neutral utterances, similarly to the TDNN NSER models, with the misclassification of neutral utterances being more apparent when the CNN+LSTM NSER model uses Mel-spectrogram as input. Fear and surprise utterances were classified accurately as well, with both being up-sampled labels.

In comparison with Zhao, Mao and Chen’s work, which features the usage of Local Feature Learning Blocks (LFLB) intended as a substitute for CNN blocks, with each LFLB block containing a convolution, batch normalization, exponential linear unit, and a max-pooling layer in sequence. The model architecture itself includes 4 LFLB blocks followed by an LSTM layer used for persistent dependencies as well as a fully connected layer. The model uses the IEMOCAP dataset, using mel-spectrogram as input and being able to classify 6 different emotion labels, namely Angry, Excited, Frustrated, Happy, Neutral, and Sad. The model by Zhao et al. was able to reach an accuracy of 62.07% (Zhao et al., 2019), which was lower than the accuracy score achieved by the CNN+LSTM NSER model when using either the MFCC or the Mel-Spectrogram feature set as input.

## 5.10 Base CNN

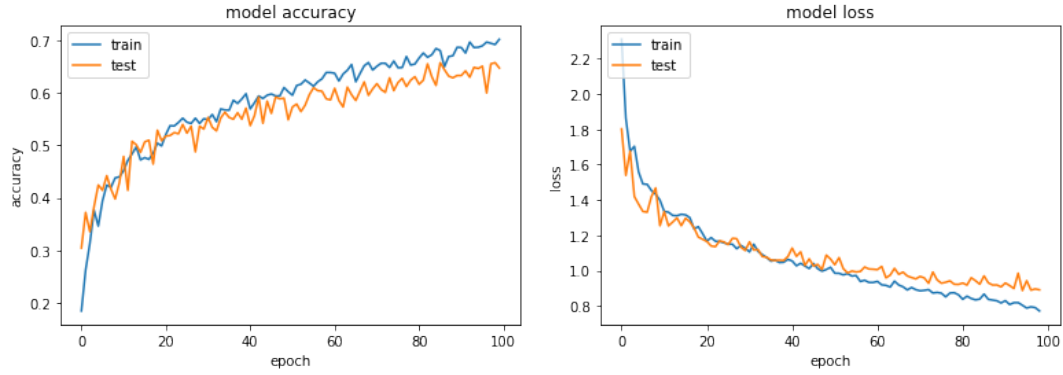


Figure 5.18: Base CNN NSER MFCC Performance Results

When using MFCC as input, Base CNN NSER was able to achieve a training accuracy of 75.13%, a validation accuracy of 69.33%, a training loss of 0.6730, a validation loss of 0.7913, a training f1-score of 0.6244 and a validation f1-score of 0.5642.

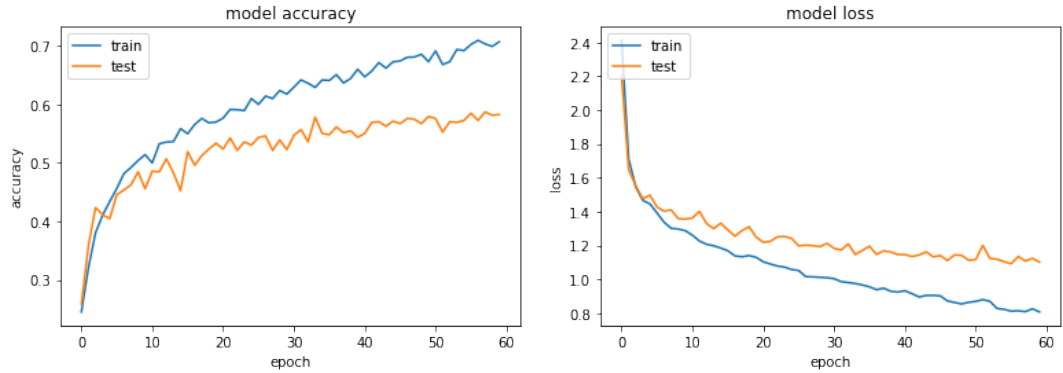


Figure 5.19: Base CNN NSER Mel-spectrogram Performance Results

As for the Mel-Spectrogram feature, the Base CNN NSER model was able to achieve achieve a training accuracy of 74.21% and a validation accuracy of 57.22%. In terms of loss, the model performed with a training loss of 0.7535 and a validation loss of 1.1422. Lastly, in terms of f1-score, the model showed a training f1-score of 0.5644 and a validation f1-score of 0.4518. The model ran for 100 epochs when using MFCC as input, while running for only 60 epochs when Mel-spectrogram was used instead. Both models had a batch size of 50 and based

from the results, the model learned best from the MFCC features similarly to both TDNN and ResNet50.

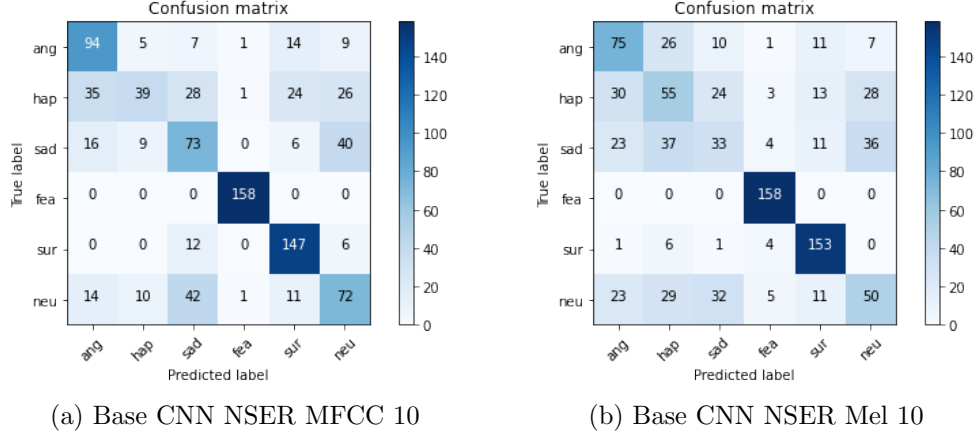


Figure 5.20: Base CNN NSER Confusion Matrices

Similar studies using CNN for speech emotion recognition include a study in 2021 by Heracleous and Yoneyama developed a convolutional neural network that uses a 12 MFCC extracted using the i-vector extraction method as input. The CNN architecture includes an input layer followed by 4 blocks, each block containing a convolution filter bank and a pooling layer, with a fully connected layer at the end. The dataset used was the entire IEMOCAP dataset, which is considered to be induced, and the model was trained and tested to classify 4 emotion labels, namely Angry, Happy, Sad, and Neutral. The model produced an accuracy of 55.2% (Heracleous & Yoneyama, 2021), which is lower compared to the 69.33% and 57.22% validation accuracy scores achieved by the Base-CNN NSER model.

Another study that uses CNN was also conducted in Chauhan et al in 2021, developing a CNN architecture that includes 5 convolution blocks, each each one containing a 2D convolution layer, batch normalization layer, Relu activation layer, and a max pooling layer. The model receives log Mel-spectrogram as input, and was trained and tested to classify 4 emotion labels, namely Angry, Happy, Sad, and Neutral. The model achieved an accuracy of 59.33% (Chauhan, Sharma, & Varma, 2021), which is lower than the 69.33% validation accuracy score of the Base CNN NSER model using MFCC as input, but higher than the 57.22% validation accuracy when Mel-spectrogram is used.

## 5.11 Further Analysis of Deep Learning Models

Further experiments were conducted using the deep learning models developed using both the data that includes all utterances equalized 29-seconds and another dataset that only equalized the utterances to 10 seconds to reduce zero-padding. These test were conducted in order to assess whether zero padding has an effect in model performance and accuracy.

Name of Model	Accuracy (10s)	Loss (10s)	F1-score (10s)	Accuracy (29s)	Loss (29s)	F1-score (29s)
TDNN (Dense) NSER	61.44%	0.9845	0.4776	60.11%	1.0200	0.5471
TDNN (Convolution) NSER	70.22%	0.7951	0.5387	63.78%	0.8872	0.4920
CNN + LSTM NSER	63.89%	0.8617	0.5388	63.44%	0.8567	0.5266
RESNET50 NSER	65.44%	2.2547	0.6460	67.44%	1.7199	0.6411
Base-CNN NSER	69.33%	0.7913	0.5642	64.78%	0.8917	0.5135

Table 5.3: Deep Learning Experiments 10s Vs 29s MFCC

When comparing the results of the models using the 29-second data and the zero-padding reduced 10 second data, the results show a similar pattern to those taken from the 10s and 29s comparison ML experiments, which show no significant difference when comparing the performance of each deep learning model using both datasets. The model to show the greatest difference between the 10s and 29s data was the TDNN with convolutional layers, which had a difference of only approximately 7%.

Name of Model	Accuracy (10s)	Loss (10s)	F1-score (10s)	Accuracy (29s)	Loss (29s)	F1-score (29s)
TDNN (Dense) NSER	64.44%	0.8370	0.5502	63.11%	0.9144	0.5070
TDNN (Convolution) NSER	62.56%	0.9737	0.5239	65.44%	0.8580	0.5339
CNN + LSTM NSER	64.00%	0.8791	0.5317	64.44%	0.8370	0.5502
RESNET50 NSER	49.89%	1.4104	0.3862	30.00%	5.0102	0.2629
Base-CNN NSER	57.22%	1.1422	0.4518	58.22%	1.1010	0.4561

Table 5.4: Deep Learning Experiments 10s Vs 29s Mel-Spectrogram

The same pattern is present when using Mel-Spectrogram as input, with little difference when comparing individual model results using the 29-second data and the zero-padding reduced 10-second data. The exception would be the 2D model RESNET50, which had its overall accuracy reduced from 57.22% to 30% when 29s data was used instead of 10s, which might be due to RESNET50 originally being designed for image recognition, which is incompatible with the IEMOCAP naturalistic data, resulting in an overall poorer performance compared to other DL methods. These findings further solidify the hypothesis of the reduction of zero padding have little to no effect on model performance.

Another experiment was conducted to provide a visual result on whether the implementation of both zero-padding and clipping the utterance will affect the result of models recognizing emotions. The percentage of recognized utterances for clipped and unclipped utterances from the 10s and 29s will be taken, in order to answer the hypothesis stated.

Emotion	Train (Clipped)	Test (Clipped)	Train (Unclipped)	Test (Unclipped)
Anger	2	8	368	122
Happy	3	9	344	144
Sad	12	10	344	134
Neutral	9	6	341	144
Surprise	6	7	329	158
Fear	0	0	342	158

Table 5.5: Number of Train and Test Data for Clipped and Unclipped

The table 5.5 refers to the amount of train and test data per emotion. The train and test data that are enumerated in the table is classified as clipped or unclipped. Clipped data refers to the data that was more than 10 seconds, in which only a portion of the audio data was used so that the length would only be 10 seconds. The Unclipped data refers to audio data that was less than 10 seconds, in which zero-padding of silence was performed until the data reaches 10 seconds for length equalization in the experiments. We observe that there are very low number of clipped data in both test and train, and that there are no fear data that was clipped.

Model	Recognized Clipped (10s)	Recognized Unclipped (29s)
TDNN (Dense) NSER	60.0%	61.5%
TDNN (Convolution) NSER	67.5%	70.4%
CNN+LSTM NSER	77.5%	63.3%
Base-CNN NSER	77.5%	69.0%

Table 5.6: Ratio of Recognized Clipped and Unclipped Test Data using MFCC as Input

For the results on MFCC as input for the deep learning models, the data shows that the clipped data had a generally higher recognition rate across all deep learning models. For CNN with LSTM NSER, as well as Base-CNN NSER, the clipped audios had a higher recognition rate than unclipped audio. However, for TDNN with Dense NSER as well as TDNN (Convolution) NSER, the unclipped



audio is recognized more. It should be noted that since there are less data for clipped audio, a small change in the amount of emotions recognized would have a more impact on the recognition rate compared to unclipped test data. This indicates that clipping audio data has no significant effect on the deep learning model’s ability to recognize emotions.

Model	Recognized Clipped (10s)	Recognized Unclipped (29s)
TDNN (Dense) NSER	67.5%	61.0%
TDNN (Convolution) NSER	60.0%	65.4%
CNN+LSTM NSER	60.0%	64.2%
Base-CNN NSER	52.5%	57.4%

Table 5.7: Ratio of Recognized Clipped and Unclipped Test Data using Mel-Spectrogram as Input

For Mel-spectrogram as input, with the exception of TDNN NSER with dense layer, unclipped data has more recognition rate when performing using the features from 10 second audio clips. The same principle applies in this case in which the small amount of clipped data is insufficient to form a conclusion that clipping data has an effect in a deep learning model’s ability to properly recognize emotions, especially with the inconsistent results wherein half of the models in the Table 5.7 show higher recognition rate for unclipped while the other half shower higher recognition rate for clipped audio. It should also be noted that there is not enough data representation for fear emotion, as there are 0 clipped audio data. As such, there is not much difference when comparing the ratio of recognized clipped and unclipped test data.

## 5.12 Discussion

Results show that for both the traditional machine learning models, using Mel-Spectrogram as input results in more consistent results than if MFCC is used as input, with all 5 machine learning methods getting higher performance metrics when using Mel-spectrogram as input, with the exception of Multi-Layer Perceptron, which performs better with MFCC as input.

For the deep learning models tested, the MFCC feature set shows a much higher performance than when Mel-spectrogram is used as input. All deep learning models used in the study achieved a higher accuracy and F1-score when using MFCC as opposed to Mel-Spectrogram, with CNN+LSTM NSER being the only model to have higher performance metrics, beating MFCC by a very small margin. The summarized results of the deep learning model performance can be found in Table 5.3 for the MFCC results and 5.4 for the Mel-Spectrogram results .

In addition, both traditional machine learning and deep learning models were able to accurately classify fear and surprise utterance, both of which were up-sampled, as well experience difficulty in classifying happy, sad utterances, which were prone to being misclassified as neutral and vice versa. It was hypothesized that some utterances belonging to both neutral and happy or sad emotion labels might be similar, and as such a Mel Spectrogram visualization of some of these utterances were viewed, and upon inspection show instances of the visual Mel-Spectrogram representation of happy or sad utterances being similar to some neutral utterances.

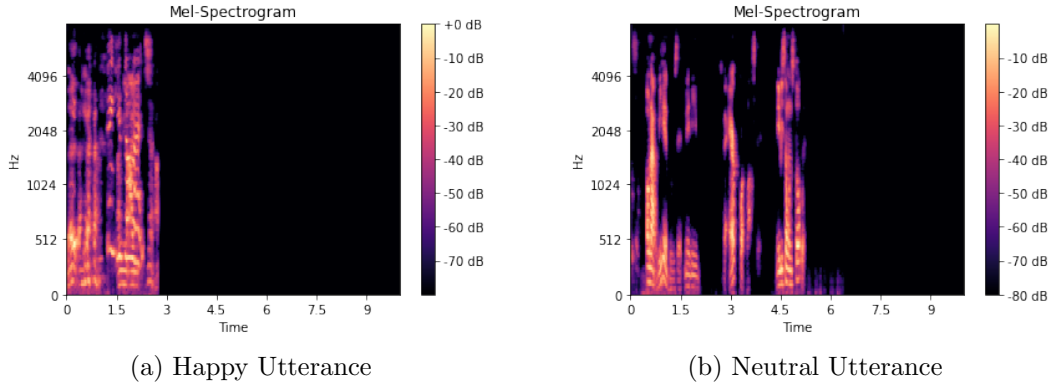


Figure 5.21: Happy Comparison with Neutral Utterance

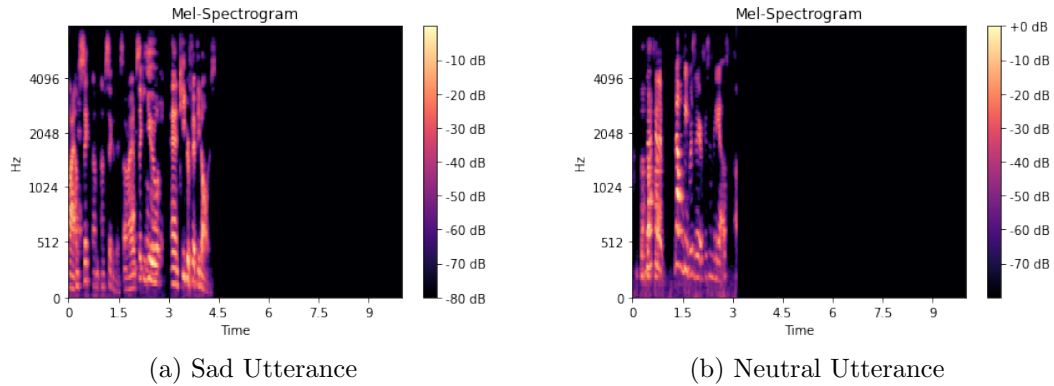


Figure 5.22: Sad Comparison with Neutral Utterance

These instances show that the similarities could be an indicator as to why the emotion labels of happy and sad tend to misclassify. Additionally, a study by Kumawat in 2021 also tested the IEMOCAP using TDNN architecture. The study also indicated the individual performance of each emotion label, and noted that the accuracy of happy utterances were at 40.05% while sad utterances performed at 41.63% when using a chunk training size of 3 chunks (Kumawat & Routray, 2021).

In another study that used Convolutional Neural Networks for speech emotion recognition using IEMOCAP as one of the dataset for experiments, showed misclassification with neutral and happy label, with 34% of neutral emotions being misclassified as happy (Heracleous & Yoneyama, 2021). Another study that also used IEMOCAP dataset with Convolutional Neural Networks with Long Short-Term Memory also showed misclassification with happy and neutral labels, with 29.03% of happy speech emotion data misclassified as neutral, when only 25.81% of happy emotion data are properly classified (Zhao et al., 2019). As observed, misclassification of labels tend to occur in speech emotion recognition studies that used the IEMOCAP dataset.

Additionally, the accuracy and F1-score metrics of the highest performing models of the Machine Learning and Deep Learning models were compared with each other, to assess whether it is preferable to use machine learning or deep learning methods when using naturalistic data from the IEMOCAP dataset.

The models to be compared will be split between the best performing models using both the MFCC and Mel-Spectrogram feature set, as well as the 10-second and 29-second dataset to check whether the reduction or addition of zero padding affects the feasibility of using deep learning over machine learning.

Name of Model	Accuracy (10s)	F1-score(10s)
Random Forest Classifier	76.6%	0.755
TDNN (Convolution) NSER	70.22%	0.5387

Table 5.8: Best DL/ML model for MFCC 10s

Name of Model	Accuracy (29s)	F1-score(29s)
Random Forest Classifier	75.6%	0.742
Base-CNN NSER	64.78%	0.5135

Table 5.9: Best DL/ML model for MFCC 29s

Based on the results, Random Forest Classifier, which was best performing machine learning model for both the 10s and 29s datasets, performed better than both the TDNN Convolution and Base-CNN, which were the highest performing deep learning models for the 10s and 29s dataset respectively. The RFC had both a higher accuracy and F1-score compared to the two deep learning models, which indicates that the usage of machine learning is more preferred over deep learning when using IEMOCAP data.

Name of Model	Accuracy (10s)	F1-score(10s)
Random Forest Classifier	77.2%	0.76
TDNN (Dense) NSER	64.44%	0.5511

Table 5.10: Best DL/ML model for Mel 10s

Name of Model	Accuracy (29s)	F1-score(29s)
Random Forest Classifier	77.7%	0.764
TDNN (Convolution) NSER	65.44%	0.5339

Table 5.11: Best DL/ML model for Mel 29s

Looking at the results with Mel-Spectrogram as input for both the 10s and 29s data, RFC is still the highest performing model, and comparing to the highest performing deep learning models for both the 10s and 29s datasets, which are the TDNN Dense and TDNN Convolutional models respectively, the machine learning model still outperforms both of them in regards to accuracy and F1-score, leading to the conclusion of machine learning models performing more accurately than deep learning ones when naturalistic IEMOCAP data is used.

# Chapter 6

## Conclusion

To conclude, based on the results achieved during experimentation, both machine learning and deep learning models that were developed were able to perform in line with present studies in terms of validation accuracy while using the improvised sessions in the IEMOCAP dataset. One other finding was the preference of traditional machine learning over deep learning methods, with machine learning models achieving higher performance in comparison with results taken from the developed deep learning models, with the exception of Support Vector Machines and Logistic Regression. It should also be noted that the models used in this experiment are not optimal models for speech emotion recognition. Since the deep learning model architectures, which are ResNet50, Time Delay Neural Network, Convolutional Neural Network with Long Short-Term Memory, which are based on other models from literature, these models, as well as the Base-CNN, have for improvement as there are architectures that has higher performance compared to the models used in the experiments.

As for the comparison of results using deep neural networks, the Time Delay Neural Network (TDNN) using convolution layers has the highest overall performance based on accuracy. Although Base-CNN NSER has a slightly higher performance than TDNN with convolution layers when using MFCC features from 29-second audio data, and TDNN with dense or fully connected layers performed slightly better in terms of accuracy when using Mel-Spectrogram features from 10-second audio data. Based on these findings, deep learning models have the highest performance when using a simpler architecture. This means that a smaller number of convolution blocks tends to result for the models to learn the audio features better, especially when using the small amount of data from IEMOCAP dataset. As an example, we see an opposite case when observing the ResNet50 model, which has a more robust architecture with significantly more convolution blocks

and layer. The ResNet50 model was not able to learn the audio features despite its architecture, regardless of using MFCC or Mel-Spectrogram, as displayed by the high value of loss across all the experiments.

Furthermore, based on the experiments comparing the results from both 10-second silence padded audio data and 29-second silence padded audio data, results show that there is no significant difference in terms of model performance in both machine learning models as well as deep learning models regardless of which features, whether MFCC or Mel-Spectrogram, were used as input. These findings indicate that reducing the amount of silence that is padded to each audio for length equalization has no significant effect on the overall model performance of both machine learning and deep learning models. Additionally, models using the improvised segment of the IEMOCAP database tend to interchangeably misclassify happy and sad utterances with neutral utterances. Further tests show that similarities are present between certain happy or sad utterances with certain neutral one. To add, there is the lack of anger, fear and surprise utterances in the improvised sessions in the IEMOCAP dataset. In this study, the utterances were upsampled, which while being able to increase the overall utterance count for both labels, resulted in models over-fitting in regards to these utterances.

Finally, when comparing clipped audio with unclipped audio based on the recognition rate of how much data was recognized per emotion label, we found that there is no significant difference in recognition rate. We concluded that this is the case because there are significantly less data for clipped audio, which means that a small change in the amount of emotions recognized would have a large impact on the recognition rate compared to unclipped test data. Another reason for this conclusion is that there is not even enough data for some emotion labels, such as *anger* and *happy*, and that there is even 0 clipped data for *fear*. Because of these factors, there is no significant difference in recognition rate between clipped and unclipped data, and there is not enough data to determine whether clipping audio data leads to higher model performance.

## 6.1 Recommendations

For the recommendation, all the models in the study could be optimized further. In regards to the machine learning models, the default parameters should be explored as it can help the models perform better in terms of accuracy and other metrics. Using grid or inform search can help in finding the right values for those parameters. However, it can be computational expensive, so it will take a large amount of time to perform. This applies to deep learning models, in

which modifying the training parameters as well as using different architectures can improve the performance of the model.

Another recommendation for naturalistic speech emotion recognition is to use multimodal or contextual information when analyzing emotions. As naturalistic emotions do not occur in a vacuum, which means that each emotion expression may vary from each speaker, using multiple modalities would be a good solution in performing emotion recognition. Possible modalities aside from speech are facial expressions, body language or gestures (Castellano, Kessous, & Caridakis, 2008). A study by Castellano et al. in 2008 explored the topic by performing emotion recognition using multiple modalities, and the results can be a reliable source for future studies that aim to perform speech emotion recognition with additional modalities.

In addition, dynamic time warping can be explored to perform length equalization for each utterance in the dataset. This can be done in place of zero padding, as zero padding affect the feature extraction by including the excess silence in the audio. Furthermore, as a solution to misclassification of emotions in the IEMOCAP dataset, re-annotation of the emotion labels on the speech data can be explored. Aside from the audio segments, video clips from the dyadic sessions are also available, and annotators can make use of the facial expressions and certain keywords to appropriately label the emotions expressed by the participants. A much thorough annotation process may contribute in properly labeled data, and consequently machine learning and deep learning models can classify emotions precisely.

Lastly, other datasets that are larger in size and have a more balanced data per emotion label can be explored instead of IEMOCAP. Especially when dealing with small data, there may not be enough utterances for any model to learn. In the case of study, the number of anger, fear and surprise utterances found in the improvised sessions in the IEMOCAP dataset had to be upsampled in order to have enough data per emotion to perform the experiments. Possible datasets include the Large-Scale Speech Emotion Dataset (LSSSED) or the Vera am Mittag audio-visual database (VAM).



# References

- Abbaschian, B. J., Sierra-Sosa, D., & Elmaghraby, A. (2021). Deep learning techniques for speech emotion recognition, from databases to models. *Sensors*, 21(4). Retrieved from <https://www.mdpi.com/1424-8220/21/4/1249> doi: 10.3390/s21041249
- Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017). Understanding of a convolutional neural network. In *2017 international conference on engineering and technology (icet)* (p. 1-6). doi: 10.1109/ICEngTechnol.2017.8308186
- Allen, J., & Rabiner, L. (1977). A unified approach to short-time fourier analysis and synthesis. *Proceedings of the IEEE*, 65(11), 1558-1564. doi: 10.1109/PROC.1977.10770
- Anwar, A. (2021). What is a convolutional neural network? In *Towards data science*. Retrieved from <https://towardsdatascience.com/a-visualization-of-the-basic-elements-of-a-convolutional-neural-network-75fea30cd78d>
- Ayadi, S., & Lachiri, Z. (2022a). Deep neural network for visual emotion recognition based on resnet50 using song-speech characteristics. In *2022 5th international conference on advanced systems and emergent technologies (ic\_aset)* (p. 363-368). doi: 10.1109/IC\_ASET53395.2022.9765898
- Ayadi, S., & Lachiri, Z. (2022b, May). *Deep neural network for visual emotion recognition based on resnet50 using song-speech characteristics*. Retrieved from <https://ieeexplore.ieee.org/abstract/document/9765898/>
- Bachorowski, J., & Owren, M. (2009). Emotion in speech. In L. R. Squire (Ed.), *Encyclopedia of neuroscience* (p. 897-901). Oxford: Academic Press. Retrieved from <https://www.sciencedirect.com/science/article/pii/B9780080450469018970> doi: <https://doi.org/10.1016/B978-008045046-9.01897-0>
- Baker, C. (2004). *Behavioral genetics: an introduction to how genes and environments interact through development to shape differences in mood, personality, and intelligence*. American Association for the Advancement of Science.
- Bestelmeyer, P. E. G., Kotz, S. A., & Belin, P. (2017). Effects of emotional

- valence and arousal on the voice perception network. *Social Cognitive and Affective Neuroscience*, 12(8), 1351–1358. doi: 10.1093/scan/nsx059
- Breazeal, C., & Aryananda, L. (2002, 01). Recognition of affective communicative intent in robot-directed speech. *Auton. Robots*, 12, 83-104. doi: 10.1023/A:1013215010749
- Brownlee, J. (2019a, Aug). *A gentle introduction to padding and stride for convolutional neural networks*. Retrieved from <https://machinelearningmastery.com/padding-and-stride-for-convolutional-neural-networks/>
- Brownlee, J. (2019b, Aug). *How to configure the learning rate when training deep learning neural networks*. Retrieved from <https://machinelearningmastery.com/learning-rate-for-deep-learning-neural-networks/>
- Brownlee, J. (2021, Oct). *How to choose an optimization algorithm*. Retrieved from <https://machinelearningmastery.com/tour-of-optimization-algorithms/>
- Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W., & Weiss, B. (2005, 01). A database of german emotional speech. In (Vol. 5, p. 1517-1520).
- Busso, C., Bulut, M., Lee, C.-C., Kazemzadeh, A., Mower Provost, E., Kim, S., ... Narayanan, S. (2008, 12). Iemocap: Interactive emotional dyadic motion capture database. *Language Resources and Evaluation*, 42, 335-359. doi: 10.1007/s10579-008-9076-6
- Cao, H., Verma, R., & Nenkova, A. (2015). Speaker-sensitive emotion recognition via ranking: Studies on acted and spontaneous speech. *Computer speech & language*, 28(1), 186-202.
- Castellano, G., Kessous, L., & Caridakis, G. (2008). Emotion recognition through multiple modalities: Face, body gesture, speech. In C. Peter & R. Beale (Eds.), *Affect and emotion in human-computer interaction: From theory to applications* (pp. 92–103). Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved from [https://doi.org/10.1007/978-3-540-85099-1\\_8](https://doi.org/10.1007/978-3-540-85099-1_8) doi: 10.1007/978-3-540-85099-1\_8
- Chae, M., Kim, T.-H., Shin, Y. H., Kim, J.-W., & Lee, S.-Y. (2018). *End-to-end multimodal emotion and gender recognition with dynamic joint loss weights*.
- Chaudhary, K. (2020). Spectrogram features for a speech recognition system.. Retrieved from <https://towardsdatascience.com/understanding-audio-data-fourier-transform-fft-spectrogram-and-speech-recognition-a4072d228520>
- Chauhan, K., Sharma, K. K., & Varma, T. (2021). Speech emotion recognition using convolution neural networks. In *2021 international conference on artificial intelligence and smart systems (icaais)* (p. 1176-1181). doi: 10.1109/ICAIS50930.2021.9395844
- Chen, C. H., & Wang, P. S.-P. (2005). *Handbook of pattern recognition and computer vision*. World Scientific.

- Chen, M., He, X., Yang, J., & Zhang, H. (2018). 3-d convolutional recurrent neural networks with attention model for speech emotion recognition. *IEEE signal processing letters*, 25(10), 1440-1444.
- Chenchah, F., & Lachiri, Z. (2014). Speech emotion recognition in acted and spontaneous context. *Procedia Computer Science*, 39, 139-145. Retrieved from <https://www.sciencedirect.com/science/article/pii/S1877050914014380> (The 6th international conference on Intelligent Human Computer Interaction, IHCI 2014) doi: <https://doi.org/10.1016/j.procs.2014.11.020>
- Chernykh, V., & Prikhodko, P. (2018). *Emotion recognition from speech with recurrent neural networks*.
- Clark, E. A., Kessinger, J., Duncan, S. E., Bell, M. A., Lahne, J., Gallagher, D. L., & Okeefe, S. F. (2020). The facial action coding system for characterization of human affective response to consumer product-based stimuli: A systematic review. *Frontiers in Psychology*, 11. doi: 10.3389/fpsyg.2020.00920
- Deb, S., & Dandapat, S. (2019). Emotion classification using segmentation of vowel-like and non-vowel-like regions. *IEEE transactions on affective computing*, 10(3), 360-373.
- De la Torre, F., Chu, W.-S., Xiong, X., Vicente, F., Ding, X., & Cohn, J. (2015, 05). Intraface.. doi: 10.1109/FG.2015.7163082
- Dellaert, F., Polzin, T., & Waibel, A. (1996). Recognizing emotion in speech. In *Proceeding of fourth international conference on spoken language processing. icslp '96* (Vol. 3, p. 1970-1973 vol.3). doi: 10.1109/ICSLP.1996.608022
- Deng, J., Xu, X., Zhang, Z., Frühholz, S., & Schuller, B. (2018). Semisupervised autoencoders for speech emotion recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(1), 31-43. doi: 10.1109/TASLP.2017.2759338
- Dhall, A., Goecke, R., Lucey, S., & Gedeon, T. (2012). Collecting large, richly annotated facial-expression databases from movies. *IEEE MultiMedia*, 19(3), 34-41. doi: 10.1109/MMUL.2012.26
- Discrete cosine transform (dct). (2006). In B. Furht (Ed.), *Encyclopedia of multimedia* (pp. 203-205). Boston, MA: Springer US. Retrieved from [https://doi.org/10.1007/0-387-30038-4\\_61](https://doi.org/10.1007/0-387-30038-4_61) doi: 10.1007/0-387-30038-4.61
- Dung, L. T. (2019). *Tddn lstm speech recognition*. Retrieved from <https://github.com/DungLuongTuan/TDNN-LSTM-Speech-Recognition>
- du Preez, S. J., Lall, M., & Sinha, S. (2009). An intelligent web-based voice chat bot. In *Ieee eurocon 2009* (p. 386-391). doi: 10.1109/EURCON.2009.5167660
- Ekman, P. (1992a). Are there basic emotions? *Psychological Review*, 99(3), 550-553. doi: 10.1037/0033-295x.99.3.550
- Ekman, P. (1992b). An argument for basic emotions. *Cognition and Emotion*, 6(3-4), 169-200. Retrieved from <https://doi.org/10.1080/>

- 02699939208411068 doi: 10.1080/02699939208411068
- Ekman, P. (1994). Strong evidence for universals in facial expressions: A reply to russell's mistaken critique. *Psychological Bulletin*, 115(2), 268-287. doi: <https://doi.org/10.1037/0033-2909.115.2.268>
- Ekman, P. (2021, May). *Paul ekman's theory of emotion*. Retrieved from <https://www.envisionyourevolution.com/evolution-emotion/paul-ekman-theory-of-emotion/2149/>
- Ekman, P., & Friesen, W. V. (1978). Facial action coding system. *PsycTESTS Dataset*. doi: 10.1037/t27734-000
- Elelu, A. (2021, Sep). *Mfcc: The dummy's guide*. Medium. Retrieved from <https://medium.com/@abdulsalamelelu/mfcc-the-dummys-guide-fd7fc471db76>
- Fan, W., Xu, X., Xing, X., Chen, W., & Huang, D. (2021). *Lssed: a large-scale dataset and benchmark for speech emotion recognition*.
- Gerczuk, M., Amiriparian, S., Ottl, S., & Schuller, B. W. (2021). Emonet: A transfer learning framework for multi-corpus speech emotion recognition. *IEEE Transactions on Affective Computing*.
- Gayayomi, M., & Bansal, A. (2016, 06). *Unifying geometric features and facial action units for improved performance of facial expression analysis*.
- Greenberg, M. (2022). *Speech emotion recognition (ser) in real-time*. Retrieved from [https://github.com/MeidanGR/SpeechEmotionRecognition\\_Realtime](https://github.com/MeidanGR/SpeechEmotionRecognition_Realtime)
- Grimm, M., Kroschel, K., & Narayanan, S. (2008). The vera am mittag german audio-visual emotional speech database. In *2008 ieee international conference on multimedia and expo* (p. 865-868). doi: 10.1109/ICME.2008.4607572
- Grossberg, S., & Levine, D. S. (1987, Dec). Neural dynamics of attentionally modulated pavlovian conditioning: blocking, interstimulus interval, and secondary reinforcement. *Appl. Opt.*, 26(23), 5015–5030. Retrieved from <http://ao.osa.org/abstract.cfm?URI=ao-26-23-5015> doi: 10.1364/AO.26.005015
- Gunes, H., & Pantic, M. (2010, 01). Automatic, dimensional and continuous emotion recognition. *IJSE*, 1, 68-99. doi: 10.4018/jse.2010101605
- He, K., Zhang, X., Ren, S., & Sun, J. (2015). *Deep residual learning for image recognition*.
- Heracleous, P., & Yoneyama, A. (2021). *A comprehensive study on bilingual and multilingual speech emotion recognition using a two-pass classification scheme*. U.S. National Library of Medicine. Retrieved from <https://pubmed.ncbi.nlm.nih.gov/31415592/>
- Huang, C., Gong, W., Fu, W., & Feng, D. (2014, 08). A research of speech emotion recognition based on deep belief network and svm. *Mathematical Problems in Engineering*, 2014, 1-7. doi: 10.1155/2014/749604

- Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3), 90–95. doi: 10.1109/MCSE.2007.55
- Ingale, A. B., & Chaudhari, D. S. (2012). *Speech emotion recognition* (Vol. 2).
- Izard, C. E. (2009). Emotion theory and research: Highlights, unanswered questions, and emerging issues. *Annual Review of Psychology*, 60(1), 1-25. Retrieved from <https://doi.org/10.1146/annurev.psych.60.110707.163539> (PMID: 18729725) doi: 10.1146/annurev.psych.60.110707.163539
- Jeon, M. (2017). Chapter 1 - emotions and affect in human factors and human-computer interaction: Taxonomy, theories, approaches, and methods. In M. Jeon (Ed.), *Emotions and affect in human factors and human-computer interaction* (p. 3-26). San Diego: Academic Press. Retrieved from <https://www.sciencedirect.com/science/article/pii/B978012801851400001X> doi: <https://doi.org/10.1016/B978-0-12-801851-4.00001-X>
- Ji, Q., Huang, J., He, W., & Sun, Y. (2019, 02). Optimized deep convolutional neural networks for identification of macular diseases from optical coherence tomography images. *Algorithms*, 12, 51. doi: 10.3390/a12030051
- Korstanje, J. (2021, Aug). *The f1 score*. Towards Data Science. Retrieved from <https://towardsdatascience.com/the-f1-score-bec2bbc38aa6>
- Kraus, M. (2017, 10). Voice-only communication enhances empathic accuracy. *American Psychologist*, 72, 644-654. doi: 10.1037/amp0000147
- Kugler, L., & Lehner, S. (2019). *Matehiw // machine learning techniques for high-impact weather*. Retrieved from [https://github.com/MATEHIW-project/matehiw-project.github.io/blob/master/assets/ESoWC2019\\_ml\\_flood\\_final\\_pres.pdf](https://github.com/MATEHIW-project/matehiw-project.github.io/blob/master/assets/ESoWC2019_ml_flood_final_pres.pdf)
- Kumawat, P., & Routray, A. (2021). Applying tdnn architectures for analyzing duration dependencies on speech emotion recognition. *Proc. Interspeech 2021*, 3410–3414.
- Lazarus, R. S., & Folkman, S. (1984). *Stress, appraisal, and coping*. Springer.
- LeDoux, J. E., & Brown, R. (2017). A higher-order theory of emotional consciousness. *Proceedings of the National Academy of Sciences*, 114(10), E2016–E2025. Retrieved from <https://www.pnas.org/content/114/10/E2016> doi: 10.1073/pnas.1619316114
- Li, K. (2021, Oct). *How to choose a learning rate scheduler for neural networks*. Retrieved from <https://neptune.ai/blog/how-to-choose-a-learning-rate-scheduler>
- Lim, W., Jang, D., & Lee, T. (2016). Speech emotion recognition using convolutional and recurrent neural networks. In *2016 asia-pacific signal and information processing association annual summit and conference (apsipa)* (p. 1-4). Asia Pacific Signal and Information Processing Association.
- Liu, J., Tong, J., Han, J., Yang, F., & Chen, S. (2013). Affective computing

- applications in distance education. *Proceedings of the 2013 the International Conference on Education Technology and Information Systems*. doi: 10.2991/icetis-13.2013.212
- Liu, S., Zhang, M., Fang, M., Zhao, J., Hou, K., & Hung, C.-C. (2021). Speech emotion recognition based on transfer learning from the facenet framework. *The Journal of the Acoustical Society of America*, 149(2), 1338-1345. Retrieved from <https://doi.org/10.1121/10.0003530> doi: 10.1121/10.0003530
- Livingstone, S. R., & Russo, F. A. (2018, 05). 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*, 13(5), 1-35. Retrieved from <https://doi.org/10.1371/journal.pone.0196391> doi: 10.1371/journal.pone.0196391
- Maiza, A. (2019). *The unknown benefits of using a soft-f1 loss in classification systems*. Retrieved from <https://towardsdatascience.com/the-unknown-benefits-of-using-a-soft-f1-loss-in-classification-systems-753902c0105d>
- Maklin, C. (2019). Fast fourier transform.. Retrieved from <https://towardsdatascience.com/fast-fourier-transform-937926e591cb>
- Manaswi, N. K. (2018). Understanding and working with keras. *Deep Learning with Applications Using Python*, 31-43. doi: 10.1007/978-1-4842-3516-4\_2
- Matsumoto, D., Takeuchi, S., Andayani, S., Kouznetsova, N., & Krupp, D. (1998). The contribution of individualism vs. collectivism to cross-national difference in display rules. *Asian Journal of Social Psychology*, 1, 147-165.
- McFee, B., Raffel, C., Liang, D., Ellis, D., McVicar, M., Battenberg, E., & Nieto, O. (2015). librosa: Audio and music signal analysis in python..
- Mirsamadi, S., Barsoum, E., & Zhang, C. (2017). Automatic speech emotion recognition using recurrent neural networks with local attention. In *2017 IEEE international conference on acoustics, speech and signal processing (icassp)* (p. 2227-2231). doi: 10.1109/ICASSP.2017.7952552
- Nair, P. (2018, Jul). *The dummy's guide to mfcc*. prathena. Retrieved from <https://medium.com/prathena/the-dummys-guide-to-mfcc-aceab2450fd>
- Nathanson, D. L., & Lansky, M. R. (1997).  
In *The widening scope of shame* (p. 107-138). Routledge.
- Neiberg, D., Elenius, K., Karlsson, I., & Laskowski, K. (2006). Emotion recognition in spontaneous speech. In *Proceedings of fonetik* (pp. 101-104).
- Ntalampiras, S. (2021). Speech emotion recognition via learning analogies. *Pattern Recognition Letters*, 144, 21-26. Retrieved from <https://www.sciencedirect.com/science/article/pii/S0167865521000313> doi: <https://doi.org/10.1016/j.patrec.2021.01.018>
- Padi, S., Sadjadi, S. O., Manocha, D., & Sriram, R. D. (2021). *Improved speech*

- emotion recognition using transfer learning and spectrogram augmentation.*
- Peddinti, V., Povey, D., & Khudanpur, S. (2008). A time delay neural network architecture for efficient modeling of long temporal contexts..
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.
- Perlovsky, L. (2012). Cognitive function, origin, and evolution of musical emotions. *Musicae Scientiae*, 16(2), 185-199. Retrieved from <https://doi.org/10.1177/1029864912448327> doi: 10.1177/1029864912448327
- Picard, R. W. (2000). *Affective computing*. MIT Press.
- Pichora-Fuller, M. K., & Dupuis, K. (2020). *Toronto emotional speech set (tess)*. Scholars Portal Dataverse. Retrieved from <https://doi.org/10.5683/SP2/E8H2MF> doi: 10.5683/SP2/E8H2MF
- Plutchik, R., & Kellerman, H. (1980). *Emotion: theory, research and experience. vol. 1, theories of emotion*. Academic Press.
- Plutchik's wheel of emotions: Feelings wheel six seconds.* (2021, Apr). Retrieved from <https://www.6seconds.org/2020/08/11/plutchik-wheel-emotions/>
- POSNER, J., RUSSELL, J. A., & PETERSON, B. S. (2005). The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology. *Development and Psychopathology*, 17(3), 715–734. doi: 10.1017/S0954579405050340
- Reed, C. L., Moody, E. J., Mgrublian, K., Assaad, S., Schey, A., & McIntosh, D. N. (2020). Body matters in emotion: Restricted body movement and posture affect expression and recognition of status-related emotions. *Frontiers in Psychology*, 11, 1961. Retrieved from <https://www.frontiersin.org/article/10.3389/fpsyg.2020.01961> doi: 10.3389/fpsyg.2020.01961
- Reyes, M., Meza, I., & Pineda, L. (2019, 01). Robotics facial expression of anger in collaborative human–robot interaction. *International Journal of Advanced Robotic Systems*, 16, 172988141881797. doi: 10.1177/1729881418817972
- Robert, J., Webbie, M., et al. (2018). *Pydub*. GitHub. Retrieved from <http://pydub.com/>
- Roberts, J. (2011). *Jiaaro/pydub: Manipulate audio with a simple and easy high level interface*. Retrieved from <https://github.com/jiaaro/pydub>
- Roberts, L. (2020). Understanding the mel spectrogram.. Retrieved from <https://medium.com/analytics-vidhya/understanding-the-mel-spectrogram-fca2afa2ce53>
- Roza, F. (2019). End-to-end learning, the (almost) every purpose ml method.. Retrieved from <https://towardsdatascience.com/e2e-the-every-purpose-ml-method-5d4f20dafee4>
- Rubin, D., & Talarico, J. (2009, 09). A comparison of dimensional models of emotion: Evidence from emotions, prototypical events, autobiograph-

- ical memories, and words. *Memory (Hove, England)*, 17, 802-8. doi: 10.1080/09658210903130764
- Russell, J. A. (2006). Emotions are not modules1. *Canadian Journal of Philosophy Supplementary Volume*, 32, 53–71. doi: 10.1353/cjp.2007.0037
- Sabra, A. (2021). Learning from audio: The mel scale, mel spectrograms, and mel frequency cepstral coefficients.. Retrieved from <https://towardsdatascience.com/learning-from-audio-the-mel-scale-mel-spectrograms-and-mel-frequency-cepstral-coefficients-f5752b6324a8>
- Sahu, G. (2019). *Multimodal speech emotion recognition and ambiguity resolution*. arXiv. Retrieved from <https://arxiv.org/abs/1904.06022> doi: 10.48550/ARXIV.1904.06022
- Sainath, T. (2020). End-to-end speech recognition..
- Sakar, D. (2018). A comprehensive hands-on guide to transfer learning with real-world applications in deep learning.. Retrieved from <https://towardsdatascience.com/a-comprehensive-hands-on-guide-to-transfer-learning-with-real-world-applications-in-deep-learning-212bf3b2f27a>
- Sander, D., & Scherer, K. R. (2009). *The oxford companion to emotion and the affective sciences*. Oxford University Press.
- Sarma, M., Ghahremani, P., Povey, D., Goel, N. K., Sarma, K. K., & Dehak, N. (2018). *Emotion identification from raw speech signals using dnns*. Retrieved from [https://danielpovey.com/files/2018\\_interspeech\\_emotion\\_id.pdf](https://danielpovey.com/files/2018_interspeech_emotion_id.pdf)
- Schoeller, F., Bertrand, P., Gerry, L. J., Jain, A., Horowitz, A. H., & Zenasni, F. (2019). Combining virtual reality and biofeedback to foster empathic abilities in humans. *Frontiers in Psychology*, 9, 2741. Retrieved from <https://www.frontiersin.org/article/10.3389/fpsyg.2018.02741> doi: 10.3389/fpsyg.2018.02741
- Schroff, F., Kalenichenko, D., & Philbin, J. (2015, Jun). Facenet: A unified embedding for face recognition and clustering. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Retrieved from <http://dx.doi.org/10.1109/CVPR.2015.7298682> doi: 10.1109/cvpr.2015.7298682
- Schuller, B., Steidl, S., Batliner, A., Vinciarelli, A., Scherer, K., Ringeval, F., ... et al. (2013, Jan). [pdf] *the interspeech 2013 computational paralinguistics challenge: social signals, conflict, emotion, autism: Semantic scholar*. Retrieved from <https://www.semanticscholar.org/paper/The-INTERSPEECH-2013-computational-paralinguistics-Schuller-Steidl/a55a7161917ff04328dc1719de85e0c4e2504559>
- Schuller, B. W. (2018, April). Speech emotion recognition: Two decades in a nutshell, benchmarks, and ongoing trends. *Commun. ACM*, 61(5), 90–99. Re-



- trieved from <https://doi.org/10.1145/3129340> doi: 10.1145/3129340
- Sebe, N., Science, F. o., Cohen, I., Labs, H., Huang, T. S., & Institute, B. (2005). *Multimodal emotion recognition*. Retrieved from [https://www.worldscientific.com/doi/abs/10.1142/9789812775320\\_0021](https://www.worldscientific.com/doi/abs/10.1142/9789812775320_0021)
- Selig, J. (2022, Apr). *What is machine learning? a definition*. Retrieved from <https://www.expert.ai/blog/machine-learning-definition/>
- Sezgin, M. C., Gunsel, B., & Kurt, G. K. (2012). Perceptual audio features for emotion detection. *EURASIP Journal on Audio, Speech, and Music Processing*, 2012(1), 1-21.
- Sharma, S. (2021, Jul). *Activation functions in neural networks*. Towards Data Science. Retrieved from <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>
- Shpigler, H. Y., Saul, M. C., Corona, F., Block, L., Cash Ahmed, A., Zhao, S. D., & Robinson, G. E. (2017). Deep evolutionary conservation of autism-related genes. *Proceedings of the National Academy of Sciences*, 114(36), 9653–9658. Retrieved from <https://www.pnas.org/content/114/36/9653> doi: 10.1073/pnas.1708127114
- Stewart, M. (2019, Feb). *Simple introduction to convolutional neural networks — by ...* Retrieved from <https://towardsdatascience.com/simple-introduction-to-convolutional-neural-networks-cdf8d3077bac>
- Summers, N. (2020). *What happened to the doughnut-shaped olly speaker*.
- Sun, T.-W. (2020). End-to-end speech emotion recognition with gender information. *IEEE Access*, 8, 152423-152438. doi: 10.1109/ACCESS.2020.3017462
- Tao, J., Liu, F., Zhang, M., & Jia, H. (2008). Design of speech corpus for mandarin text to speech..
- Tchistiakova, S. (2019, Nov). *Time delay neural network*. Retrieved from <https://kaleidoescape.github.io/tdnn/>
- Trigeorgis, G., Ringeval, F., Brueckner, R., Marchi, E., Nicolaou, M. A., Schuller, B., & Zafeiriou, S. (2016). Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network. In *2016 ieee international conference on acoustics, speech and signal processing (icassp)* (pp. 5200–5204).
- Tzirakis, P., Nguyen, A., Zafeiriou, S., & Schuller, B. W. (2021). Speech emotion recognition using semantic information. In *Icassp 2021 - 2021 ieee international conference on acoustics, speech and signal processing (icassp)* (p. 6279-6283). doi: 10.1109/ICASSP39728.2021.9414866
- Tzirakis, P., Trigeorgis, G., Nicolaou, M. A., Schuller, B. W., & Zafeiriou, S. (2017). End-to-end multimodal emotion recognition using deep neural networks. *IEEE Journal of Selected Topics in Signal Processing*, 11(8), 1301-1309. doi: 10.1109/JSTSP.2017.2764438
- Van Rossum, G., & Drake, F. L. (2009). *Python 3 reference manual*. Scotts Valley, CA: CreateSpace.

- Venkataramanan, K., & Rajamohan, H. R. (2019). *Emotion recognition from speech*.
- Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., ... SciPy 1.0 Contributors (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17, 261–272. doi: 10.1038/s41592-019-0686-2
- Wang, S., & Li, G. (2019, 04). Overview of end-to-end speech recognition. *Journal of Physics: Conference Series*, 1187, 052068. doi: 10.1088/1742-6596/1187/5/052068
- Watson, D., & Tellegen, A. (1985). Toward a consensual structure of mood. *Psychological Bulletin*, 98(2), 219–235. Retrieved from <https://doi.org/10.1037/0033-2909.98.2.219> doi: 10.1037/0033-2909.98.2.219
- Wu, S., Falk, T. H., & Chan, W.-Y. (2011). Automatic speech emotion recognition using modulation spectral features. *Speech Communication*, 53(5), 768–785. Retrieved from <https://www.sciencedirect.com/science/article/pii/S0167639310001470> (Perceptual and Statistical Audition) doi: <https://doi.org/10.1016/j.specom.2010.08.013>
- Yiu, T. (2021, Sep). *Understanding random forest*. Towards Data Science. Retrieved from <https://towardsdatascience.com/understanding-random-forest-58381e0602d2>
- Yu, Q., Maddah-Ali, M. A., & Avestimehr, A. S. (2017). Coded fourier transform. *CoRR*, abs/1710.06471. Retrieved from <http://arxiv.org/abs/1710.06471>
- Zhang, M., Yu, L., Zhang, K., Du, B., Chen, S., Jiang, X., ... Luo, W. (2020). Kinematic dataset of actors expressing emotions. *Scientific Data*, 7(1). doi: DOI:10.1038/s41597-020-00635-7
- Zhang, S., Zhang, S., Huang, T., & Gao, W. (2018). Speech emotion recognition using deep convolutional neural network and discriminant temporal pyramid matching. *IEEE transactions on multimedia*, 20(6), 1576–1590.
- Zhao, J., Mao, X., & Chen, L. (2019). Speech emotion recognition using deep 1d & 2d cnn lstm networks. *Biomedical signal processing and control*, 47, 312–323.
- Zhou, S., & Beigi, H. (2020). *A transfer learning method for speech emotion recognition from automatic speech recognition*.
- Zoltan, C. (2022, Feb). *Svm and kernel svm*. Towards Data Science. Retrieved from <https://towardsdatascience.com/svm-and-kernel-svm-fed02bef1200>

# Appendix A

## Research Ethics Documents

 De La Salle University	<b>Research Ethics Review Committee</b> Research Ethics Office, 3F Henry Sy Sr. Hall De La Salle University Manila 2401 Taft Avenue, Manila 1004, Philippines REO@dlsu.edu.ph (632) 524-4611 loc. 513	SOP No.: 2
		Form No.: 2(D)
		Version No.: 1
		Version Date: July 2016

<b>DE LA SALLE UNIVERSITY</b> <b>General Research Ethics Checklist</b>
<p><i>This checklist is to ensure that the research conducted by the faculty members and students of De La Salle University is carried out according to the guiding principles outlined in the Code of Research Ethics of the University. The investigator is advised to refer to the <u>De La Salle University Code of Research Ethics and Guide to Responsible Conduct of Research</u> before completing this checklist. Statements pertinent to ethical issues in research should be addressed below. The checklist will help the researcher/s and advisers/readers/evaluators determine whether procedures should be undertaken during the course of the research to maintain ethical standards. The University's <u>Guide to the Responsible Conduct of Research</u> provides details on these appropriate procedures.</i></p>

Researcher Details	
Students	Chuan-chen Chu Reynaldo K. Delima Jr. Nilo Cantil K. Jatiko II Jedwig Siegfried S. Tan
Thesis Adviser	Dr. Merlin Teodosia C. Suarez
Department/College	Software Technology Department/ College of Computer Studies
Proposed Title of the Research	Using Convolutional Neural Network for Naturalistic Speech Emotion Recognition (N-SER)
Term(s) and academic year in which research project is to be undertaken	Term 2 of AY 2020-2021 to Term 2 of AY 2021-2022

<p><i>This checklist must be completed AFTER the De La Salle University Code of Ethics has been read and BEFORE gathering data.</i></p>		
Questions	Yes	No
1. Does your research involve human participants (this includes new data gathered or using pre-existing data)? If your answer is <b>yes</b> , please answer <b>Checklist A (Human Participants)</b> .  Please specify if the kind of research you will be conducting falls under any of the following Human Participants sub-categories:		✓
1.A. Will you be conducting Action Research in an existing business, company, or school? If your answer is <b>yes</b> , please answer <b>Checklist F (Action Research)</b> .		✓

 De La Salle University	<h2>Research Ethics Review Committee</h2> <p>Research Ethics Office, 3F Henry Sy Sr. Hall  De La Salle University Manila  2401 Taft Avenue, Manila 1004, Philippines  REO@dlsu.edu.ph (632) 524-4611 loc. 513</p>	SOP No.: 2
		Form No.: 2(D)
		Version No.: 1
		Version Date: July 2016

1.B. Does your research involve online communities (this includes culling data from social media platforms, online forums and blogs)? If your answer is <b>yes</b> , please answer <b>Checklist G (Internet Research)</b> .		✓
1.C. Does your research involve human participants who are situated in a community and may necessitate permission to acquire access to them? If your answer is <b>yes</b> , please answer <b>Checklist H (Community Research)</b> .		✓
2. Will your research make use of documents which are not in the public domain and, thus, require permission for use from the custodian of such documents?  <b>If YES, please provide certification that permission from the custodian of the data was sought and granted.</b>	✓	
3. Will your research make use of secondary data (e.g., surveys, inventories, plans, official documents, etc.) from an institution, organization, or agency, which are not in the public domain and, thus, require permission for use from the custodian of such documents?  <b>If YES, please provide certification that permission to use the data was sought from the institution, organization, or agency and approval was granted.</b>		✓
4. Does your research involve animals (non-human subjects)? If your answer is <b>yes</b> , please answer <b>Checklist B (Animal Subjects)</b> .		✓
5. Does your research involve Wildlife? If your answer is <b>yes</b> , please answer <b>Checklist C (Wildlife)</b> .		✓
6. Does your research involve microorganisms that are infectious, disease causing or harmful to health? If your answer is <b>yes</b> , please answer <b>Checklist D (Infectious Agents)</b> .		✓
7. Does your research involve toxic/chemicals/ substances/materials? If your answer is <b>yes</b> , please answer <b>Checklist E (Toxic Agents)</b> .		✓

### Research with Ethical Issues to address:

If you have a YES answer to any of the above categories, you will be required to complete a detailed checklist for that particular category. A YES answer does not mean the disapproval of your research proposal. By providing you with a more detailed checklist, we ensure that the ethical concerns are identified so these can be addressed in adherence to the University Code of Ethics.

 De La Salle University	<b>Research Ethics Review Committee</b> Research Ethics Office, 3F Henry Sy Sr. Hall De La Salle University Manila 2401 Taft Avenue, Manila 1004, Philippines REO@dlsu.edu.ph (632) 524-4611 loc. 513	SOP No.: 2 Form No.: 2(D) Version No.: 1 Version Date: July 2016
---	---	--

### Declaration of Conflict of Interest

☒ 1. I do not have a conflict of interest in any form (personal, financial, proprietary, or professional) with the sponsor/grant-giving organization, the study, the co-investigators/personnel, or the site.

☐ 2. I do have a conflict of interest, specifically:

☐ A. I have a personal/family or professional interest in the results of the study (family members who are co-proponents or personnel in the study, membership in relevant professional associations/organizations).

Please describe the personal/family or professional interest:

☐ B. I have proprietary interest vested in this proposal (with the intent to apply for a patent, trademark, copyright, or license)

Please describe proprietary interest:

☐ C. I have significant financial interest vested in this proposal (remuneration that exceeds P250,000.00 each year or equity interest in the form of stock, stock options or other ownership interests).

Please describe financial interest:

## Large-Scale Dataset For Speech Emotion User Agreement

---

(Academic, non-commercial, not-for-profit licence)

Copyright (c) 2021 Weiquan Fan, Xiangmin Xu, Xiaofen Xing, Weidong Chen, Dongyan Huang All rights reserved.

The goal of the Large-Scale Dataset for Speech Emotion database is to develop new techniques, technology, and algorithms for automatic speech emotion analysis and recognition. The licensors are involved in an ongoing effort to develop this dataset of emotional speech. The dataset is meant to aid research efforts in the general area of developing, testing and evaluating algorithms for human speech analysis.

To advance the state-of-the-art in speech emotion recognition, this dataset is made available to the research community. Due to copyright reasons, we provide a variety of feature sets of this database (including IS13\_ComParE, spectrogram, vq-wav2vec) and corresponding label files. To receive a copy of the dataset, the requestor must agree to observe the conditions listed below.

Use is permitted in source and binary form, provided that the following conditions are met:

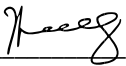
- ¼ The database is provided under the terms of this license strictly for academic, noncommercial, not-for-profit purposes.
- ¼ Redistribution, republishing, or dissemination in any form, source or binary, is not permitted without prior written approval by the licensors.
- ¼ The names of the licensors may not be used to endorse or promote products derived from this software without specific prior written permission.
- ¼ The licensors reserve the right to modify the data/license at any point. Modification of the database by licensees are not permitted.
- ¼ In no case should the still frames or sub-audios be used in any way that could cause the original subject embarrassment or mental anguish.
- ¼ Any publications arising from the use of this software, including but not limited to academic journal and conference publications, technical reports and manuals, must cite the following work:

Fan W, Xu X, Xing X, et al. LSSED: a large-scale dataset and benchmark for speech emotion recognition[C]//ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2021: 641-645.

THIS DATABASE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO

EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS DATABASE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. THE PROVIDER OF THE DATABASE MAKES NO REPRESENTATIONS AND EXTENDS NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED. THERE ARE NO EXPRESS OR IMPLIED WARRANTIES THAT THE USE OF THE MATERIAL WILL NOT INFRINGE ANY PATENT, COPYRIGHT, TRADEMARK, OR OTHER PROPRIETARY RIGHTS.

If you have read and understood the user agreement and will comply with it.

Signed  \_\_\_\_\_  
Print Name Merlin Teodosia Suarez, PHD  
Institution Name De La Salle University  
Date July 6, 2021

Addition Researcher 1 Chuan, Chen-Chu  
Addition Researcher 2 Delima, Reynaldo  
Addition Researcher 3 Jatico, Nilo Kim II  
Addition Researcher 4 Tan, Jedwig Siegfried  
Addition Researcher 5 \_\_\_\_\_  
Addition Researcher 6 \_\_\_\_\_  
Addition Researcher 7 \_\_\_\_\_  
Addition Researcher 8 \_\_\_\_\_



The Signal Analysis and Interpretation Laboratory (hereinafter referred to as "SAIL") at the University of Southern California (hereinafter referred to as "LICENSOR") agree to provide to CeHCI, De La Salle University - Manila (hereinafter referred to as "LICENSEE"), a non-exclusive, non-transferable, first-party license, of the following software/data:

**Database/Software:** IEMOCAP Database,

Hereinafter referred to as the "LICENSED SOFTWARE/DATA".

All rights remain with the LICENSOR, SAIL Laboratory at the University of Southern California, who extend the license subject to the following terms and conditions:

1. LICENSEE agrees and understands that said LICENSED SOFTWARE/DATA is for internal research purposes only.
2. LICENSEE agrees not to pass said LICENSED SOFTWARE/DATA or derivatives thereof on to others without permission of the LICENSOR. LICENSEE will further make every effort to ensure that any member of his laboratory or any other person in his laboratory who is permitted access to said LICENSED SOFTWARE/DATA will also read, sign and abide by the conditions set forth in this agreement.
3. LICENSEE will not use or exploit said LICENSED SOFTWARE/DATA or derivatives thereof for any commercial purposes.
4. LICENSEE agrees to share results, bugs, uncovered deficiencies based on LICENSED SOFTWARE/DATA with the LICENSOR. LICENSEE further agrees to share direct improvements or enhancements to said LICENSED SOFTWARE/DATA with LICENSOR free of charge.
5. LICENSEE will reference LICENSED SOFTWARE/DATA appropriately, listing the LICENSED SOFTWARE/DATA and LICENSOR, in any publications involving said LICENSED SOFTWARE/DATA and whenever used, shown or referenced in public.
6. LICENSEE agrees to consult and discuss any planned performance measures, evaluations or comparisons based on LICENSED SOFTWARE/DATA with LICENSOR, prior to such activity and prior to public reporting of such experiments. This is to allow LICENSOR to confirm and potentially assist in ensuring accuracy and proper usage of LICENSED SOFTWARE/DATA. LICENSOR will make a reasonable effort to assist LICENSEE in any such activity.
7. Any data, software, information, materials or services are furnished by LICENSOR on an "as is" basis. LICENSOR does not promise nor is required to provide maintenance, installation, help or guidance.
8. WARRANTIES: LICENSOR MAKES NO WARRANTIES OR ANY KIND, EITHER EXPRESSED OR IMPLIED AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO,

WARRANTY OF FITNESS FOR PURPOSE, OR MERCHANTABILITY, EXCLUSIVITY OR RESULTS OBTAINED FROM USE OF LICENSED SOFTWARE/DATA, NOR SHALL EITHER PARTY HERETO BE LIABLE TO THE OTHER FOR INDIRECT, SPECIAL, OR CONSEQUENTIAL DAMAGES SUCH AS LOSS OF PROFITS OR INABILITY TO USE LICENSED SOFTWARE/DATA OR ANY APPLICATIONS AND DERIVATIONS THEREOF. LICENSOR DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT, AND DOES NOT ASSUME ANY LIABILITY HEREUNDER FOR ANY INFRINGEMENT OF ANY PATENT, TRADEMARK, OR COPYRIGHT ARISING FROM THE USE OF LICENSED SOFTWARE/DATA. LICENSEE AGREES THAT IT WILL NOT MAKE ANY WARRANTY ON BEHALF OF LICENSOR, EXPRESSED OR IMPLIED, TO ANY PERSON CONCERNING THE APPLICATION OF OR THE RESULTS TO BE OBTAINED WITH THE SOFTWARE/DATA UNDER THIS AGREEMENT.

9. Indemnification: LICENSEE agrees to defend, indemnify and hold harmless LICENSOR, its trustees, officers, employees, attorneys and agents from all claims or demands made against them (and any related losses, expenses or costs) arising out of or relating to LICENSEE'S and/or its SUBLICENSEES' use of, disposition of, or conduct regarding the LICENSED SOFTWARE/DATA including but not limited to, any claims of product liability, personal injury, including, but not limited to, death, damage to property or violation of any laws or regulations including, but not limited to claims of active or passive negligence.

10. LICENSEE agrees to do all things necessary to comply with the Regulations of the United States Department of Commerce relating to the Export of Technical Data, insofar as they relate to the LICENSED SOFTWARE/DATA, and to obtain the required government documents and approvals prior to the export of any technical data disclosed or the direct product related thereto.

11. As regards this Agreement, the LICENSEE agrees to be bound by the laws of the State of California. Any dispute or claim arising out of or relating to this Agreement will be settled by arbitration in Los Angeles, California in accordance with the Rules of the American Arbitration Association and judgment upon the award rendered by the arbitrator(s) may be entered in any court having jurisdiction.

As a concurrence with terms and conditions set forth above, please have an official of your company sign and date the enclosed copies of the letter. Return one copy to the undersigned and keep the other for your records.

LICENSOR:

READ AND APPROVED - LICENSEE

---

Shrikanth Narayanan

Andrew J. Viterbi Professor of  
Engineering

Director, SAIL, University of Southern  
California



---

De La Salle University - Manila

Signer represents and warrants to USC  
that s/he has authority to bind the  
LICENSEE to this agreement.

Additional signatures:

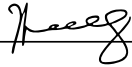
\_\_\_\_\_.

Dr. Shrikanth Narayanan

Director, SAIL

Ming Hsieh Department of Electrical  
Engineering


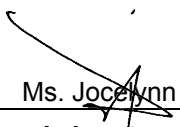
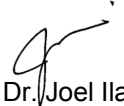
University of Southern California


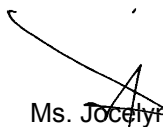

\_\_\_\_\_ 

Dr. Merlin Teodosia C. Suarez

Center for Human-Computer Innovations

De La Salle University - Manila

RESEARCH ETHICS CLEARANCE FORM For Thesis Proposals <sup>1</sup>	
<b>Names of student researcher/s :</b>	Chuan-chen Chu Reynaldo K. Delima Jr. Nilo Cantil K. Jatiko II Jedwig Siegfried S. Tan
<b>College:</b>	College of Computer Studies
<b>Department:</b>	Software Technology Department
<b>Research Title:</b>	Using Convolutional Neural Networks for Naturalistic Speech Emotion Recognition (N-SER)
<b>Course:</b>	BS Computer Science with Specialization in Software Technology
<b>Expected duration of project:</b>	from: AY20-21 Term 2 to: AY21-22 Term 2
<b>Ethical considerations</b> <ul style="list-style-type: none"> <li>The use of the Large-Scale Dataset for Speech Emotion is intended for non-profit and academic use.</li> </ul>	
<p><b>To the best of our knowledge, the ethical issues listed above have been addressed in the research.</b></p> <div style="text-align: center;">         Merlin Teodosia C. Suarez     </div> <hr/> <p><b>Name and signature of adviser/mentor</b>  <b>Date:</b></p> <div style="display: flex; justify-content: space-between; margin-top: 100px;"> <div style="width: 45%;"> <div style="text-align: center;">         Ms. Jocelyn Cu         </div> <hr/> <p><b>Name and signature of panelist</b>  <b>Date:</b></p> </div> <div style="width: 45%;"> <div style="text-align: center;">         Dr. Joel Ilao         </div> <hr/> <p><b>Name and signature of panelist</b>  <b>Date:</b> September 22, 2021</p> </div> </div>	

RESEARCH ETHICS CLEARANCE FORM For Final Thesis <sup>1</sup>	
<b>Names of student researcher/s :</b>	Chuan-chen Chu Reynaldo K. Delima Jr. Nilo Cantil K. Jatiko II Jedwig Siegfried S. Tan
<b>College:</b>	College of Computer Studies
<b>Department:</b>	Software Technology Department
<b>Research Title:</b>	Using Convolutional Neural Networks for Naturalistic Speech Emotion Recognition (N-SER)
<b>Course:</b>	BS Computer Science with Specialization in Software Technology
<b>Expected duration of project:</b>	from: AY 20-21 Term 2 to: AY 21-22 Term 2
<b>Ethical considerations</b> <ul style="list-style-type: none"> <li>The use of the Interactive Emotional Dyadic Motion Capture (IEMOCAP) is intended for non-profit and academic use.</li> </ul>	
<p><b>To the best of our knowledge, the ethical issues listed above have been addressed in the research.</b></p> <div style="text-align: center;">         _____  <b>Name and signature of adviser/mentor</b>  <b>Date:</b> </div> <div style="display: flex; justify-content: space-between; margin-top: 100px;"> <div style="width: 45%;">         _____  <b>Name and signature of panelist</b>  <b>Date:</b> </div> <div style="width: 45%;">         _____  <b>Name and signature of panelist</b>  <b>Date:</b> </div> </div>	

## Appendix B

### Similarity Report

PAPER NAME

**[THS-ST3] Thesis Document Final Revisions.pdf**

AUTHOR

-

WORD COUNT

**32442 Words**

CHARACTER COUNT

**171961 Characters**

PAGE COUNT

**111 Pages**

FILE SIZE

**2.9MB**

SUBMISSION DATE

**Jul 2, 2022 2:19 PM GMT+8**

REPORT DATE

**Jul 2, 2022 2:32 PM GMT+8****● 19% Overall Similarity**

The combined total of all matches, including overlapping sources, for each database.

- 10% Internet database
- 10% Publications database
- Crossref database
- Crossref Posted Content database
- 13% Submitted Works database

**● Excluded from Similarity Report**

- Bibliographic material



## Appendix C

### Summary of Models For Literature

Table C.1: Summary of Models.

<b>Title</b>	<b>Dataset Used</b>	<b>Audio Features</b>	<b>Algorithm</b>	<b>Accuracy</b>	<b>Possible Improvements</b>
End-to-end Multimodal Emotion and Gender Recognition with Dynamic Loss Joint Weights (2018)	IEMOCAP	Raw audio signals and video	Residual Convolutional Neural Network	69.90%	Other modalities could be explored with modalities such as age being an option taken into account
Emotion Recognition from Speech (2019)	RAVDESS	MFCC	CNN + LSTM w/ Pure Audio 1D 12 classes	61.6%	Use Atrous Spatial Pyramid Pooling to learn features better
		Log-mel gram Spectrogram	CNN + Log-mel 4-layer 12 classes  CNN + 29 Coefficients: MFCC + Delta 1-layer 2D 12 classes	70.31%  56%	Use a hierarchical structure for gender and age-groups for improvement in performance

Continuation of Table C.1					
Title	Dataset Used	Audio Features	Algorithm	Accuracy	Possible Improvements
			HMM + Log-mel Spectrogram 12 classes	-	
			CNN + Log-mel Spectrogram 3-layer 3D 12 classes	66%	
			CNN + Log-mel Spectrogram 2D w/ Global Avg. Pool 14 classes	70%	
			CNN + Log-mel Spectrogram 2D w/ Global Avg. Pool 2 classes	90%	

Continuation of Table C.1					
Title	Dataset Used	Audio Features	Algorithm	Accuracy	Possible Improvements
	TESS		CNN + LSTM Pure Audio 1D 12 classes	48.8%	
			CNN + Log-mel Spectrogram 4-layer 2D 12 classes	65%	
			CNN + 29 Coefficients: MFCC + Delta 1-layer 2D 12 classes	53%	
			HMM + Log-mel Spectrogram 12 classes	31.25%	
			CNN + Log-mel Spectrogram 3-layer 3D 12 classes	55%	

Continuation of Table C.1					
Title	Dataset Used	Audio Features	Algorithm	Accuracy	Possible Improvements
A Transfer Learning Method for Speech Emotion Recognition from Automatic Speech Recognition (2020)	IEMOCAP	MFCC and iVector	CNN + Log-mel Spectrogram 2D w/ Global Avg. Pool 14 classes	66%	Addition of another modality, specifically on semantic feature
			CNN + Log-mel Spectrogram 2D w/ Global Avg. Pool 2 classes	86%	
			Time Delay Neural Network Pre-final Bottleneck Layer	63.4%	

Continuation of Table C.1					
Title	Dataset Used	Audio Features	Algorithm	Accuracy	Possible Improvements
End-to-end Speech Emotion Recognition with Gender Information (2020)	CAISA	Uses raw audio files formatted in .wav	Time Delay Neural Net-work Bottleneck Layer	66.7%	Ability to work in real life can be looked upon
			Time Delay Neural Net-work Bottleneck Layer	69.3%	
			ASR Transfer Learning	71.7%	
			Residual Convolutional Neural Network	81.40%	

Continuation of Table C.1					
Title	Dataset Used	Audio Features	Algorithm	Accuracy	Possible Improvements
	EMODB		Residual Convolutional Neural Network with Gender In-formation Block	84.6%	In combination with R-CNN, other modalities can be further investigated without the need to classify speech data
			Residual Convolutional Neural Network	90.3%	Reduction of computational expense
			Residual Convolutional Neural Network with Gender In-formation Block	89.8%	

Continuation of Table C.1					
Title	Dataset Used	Audio Features	Algorithm	Accuracy	Possible Improvements
	IEMOCAP		Residual Convolutional Neural Network with Gender Information Block	69.9%	
Speech emotion recognition via learning analogies (2021)	EMODB	Log-mel gram	Siamese Neural Network	82.1%	Add more classes in dictionary to find similarities and dissimilarities better
		Temporal Modulation			Investigate sufficient conditions in training set
					Add more quantity in training set
Speech Emotion Recognition Using Semantic Information (2021)	SEWA	Speech2Vec	LSTM + CNN (Arousal)	Concordance Correlation Coefficient: 0.429 $\rho c$	Applying the principle of the multimodal framework in a single end-to-end model helps in simplifying the model



Continuation of Table C.1					
Title	Dataset Used	Audio Features	Algorithm	Accuracy	Possible Improvements
Adieu features? End-to-end speech emotion recognition using a deep convolutional recurrent network (2016)	RECOLA	Word2Vec	LSTM + CNN (Valence)	Concordance Correlation Coefficient: $0.503\rho c$	Study the statistics of gate activations in a network applied on an unseen speech recording
			LSTM + CNN (Liking)	Concordance Correlation Coefficient: $0.312\rho c$	
			CNN + 2 bi-directional LSTM (Arousal)	Concordance Correlation Coefficient: $0.686\rho c$	
			CNN + 2 bi-directional LSTM (Valence)	Concordance Correlation Coefficient: $0.261\rho c$	

Continuation of Table C.1					
Title	Dataset Used	Audio Features	Algorithm	Accuracy	Possible Improvements
Emotion Classification Using Segmentation of Vowel-Like and Non-Vowel Like Regions (2017)	EMODB	Speech signal (vowel-like region)	Statistical Learning Algorithm	85.10%	There must be further research done using the IEMOCAP dataset.
	IEMOCAP	Speech signal (non-vowel-like region)	Statistical Learning Algorithm	64.2%	There must be further research done with the leave-one-speaker-out evaluation with the FAU AIBO dataset.
	FAU AIBO		Statistical Learning Algorithm (Leave-one-speaker-out cross evaluation)	53.4%	

Continuation of Table C.1					
Title	Dataset Used	Audio Features	Algorithm	Accuracy	Possible Improvements
			Statistical Learning Algorithm (Pre-defined training and testing)	45.2%	
Speech Emotion Recognition using Convolutional and Recurrent Neural Networks (2016)	EMODB	Audio Signals (Short Time Fourier Transform)	CNN	87.74%	Research on concatenated CNNs
			LSTM	79.87%	
			Time Distributed CNNs	88.01%	

Continuation of Table C.1				
Title	Dataset Used	Audio Features	Algorithm	Possible Improvements
Speech Emotion Recognition Using Deep Convolutional Neural Network and Discriminant Temporal Pyramid Matching (2018)	EMODB	Mel-spectrogram	CNN	Explore the effectiveness of deep features in continuous dimension emotion recognition.
	RML		CNN	75.20%
	eNTERFACE05		CNN	79.40%
	BAUM-1s		CNN	44.03%

Continuation of Table C.1					
Title	Dataset Used	Audio Features	Algorithm	Accuracy	Possible Improvements
3-D Convolutional Recurrent Neural Networks With Attention Model for Speech Emotion Recognition (2018)	EMODB	Log-mel Spectrograms	3-D CNN + LSTM + Attention Layer	82.82%	Find other training datasets since SER architecture depends on type and size of training data.
	IEMOCAP		3-D CNN + LSTM + Attention Layer	64.74%	
Emotion Recognition from Speech With Recurrent Neural Networks (2018)	IEMOCAP	Acoustic (MFCC, Energy) features Fourier,	Connectionist Temporal Classification - Bi-directional LSTM Neural Network	54%	Dispose the hand-crafted MFCC and find other deep learning methods such as CNN
					Use domain adaptation methods
End of Table					

## Appendix D

### Full Experiment Results of Machine Learning Models

## D.1 Random Forest Classifier Parameter Experiments

Table D.1: Results With 10 Audio Sec. Clips Using MFCC Features

Number of Trees	Number of Min. Samples Split	Training Accuracy	Training score	F1-Training	Testing Accuracy	Testing score	F1-Testing
100	2	1	1		0.746	0.733	
100	4	1	1		0.742	0.729	
100	8	1	1		0.742	0.728	
100	16	0.984	0.985		0.724	0.707	
100	32	0.911	0.912		0.706	0.687	
100	64	0.799	0.797		0.664	0.644	
100	128	0.683	0.671		0.618	0.591	
200	2	1	1		0.762	0.751	
200	4	1	1		0.759	0.748	
200	8	1	1		0.756	0.743	
200	16	0.987	0.987		0.728	0.711	
200	32	0.912	0.913		0.716	0.698	
200	64	0.808	0.805		0.678	0.654	
200	128	0.696	0.684		0.616	0.588	
400	2	1	1		0.748	0.735	
400	4	1	1		0.754	0.743	
400	8	1	1		0.757	0.744	
400	16	0.990	0.990		0.738	0.724	
400	32	0.920	0.921		0.718	0.700	
400	64	0.808	0.805		0.658	0.637	

Continuation of Table						
No. Trees	No. Samples	Training Accuracy	Training score	F1-score	Testing Accuracy	Testing score
400	128	0.701	0.688		0.612	0.583
800	2	1	1		0.761	0.750
800	4	1	1		0.766	0.755
800	8	1	1		0.757	0.744
800	16	0.986	0.986		0.736	0.721
800	32	0.914	0.914		0.721	0.704
800	64	0.804	0.802		0.670	0.649
800	128	0.697	0.684		0.619	0.589
End of Table						

Table D.2: Results With 30 Audio Sec. Clips Using MFCC Features

Number of Trees	Number of Min. Samples Split	Training Accuracy	Training score	F1-score	Testing Accuracy	Testing score	F1-score
100	2	1	1		0.750	0.737	
100	4	1	1		0.743	0.731	
100	8	1	1		0.728	0.714	
100	16	0.979	0.979		0.734	0.721	
100	32	0.912	0.913		0.702	0.684	
100	64	0.798	0.795		0.666	0.647	
100	128	0.698	0.685		0.626	0.599	
200	2	1	1		0.734	0.722	
200	4	1	1		0.752	0.741	



Continuation of Table						
No. Trees	No. Samples	Training Accuracy	Training score	F1-Training	Testing Accuracy	Testing score F1-
200	8	1	1		0.752	0.739
200	16	0.983	0.984		0.728	0.714
200	32	0.912	0.912		0.712	0.694
200	64	0.805	0.802		0.667	0.641
200	128	0.697	0.685		0.616	0.589
400	2	1	1		0.749	0.738
400	4	1	1		0.756	0.742
400	8	1	1		0.752	0.740
400	16	0.987	0.987		0.732	0.718
400	32	0.918	0.918		0.717	0.699
400	64	0.804	0.802		0.674	0.652
400	128	0.698	0.685		0.622	0.595
800	2	1	1		0.753	0.741
800	4	1	1		0.752	0.740
800	8	1	1		0.753	0.741
800	16	0.988	0.988		0.736	0.720
800	32	0.915	0.915		0.712	0.694
800	64	0.812	0.810		0.672	0.652
800	128	0.700	0.689		0.616	0.588
End of Table						

Table D.3: Results With 10 Audio Sec. Clips Using Mel-spectrogram Features

Number of Trees	Number of Min. Samples Split	Training Accuracy	Training score	F1- score	Testing Accuracy	Testing score	F1- score
100	2	1	1		0.757	0.745	
100	4	1	1		0.749	0.735	
100	8	1	1		0.761	0.747	
100	16	0.990	0.990		0.749	0.735	
100	32	0.919	0.919		0.729	0.711	
100	64	0.824	0.823		0.697	0.677	
100	128	0.742	0.737		0.664	0.641	
200	2	1	1		0.772	0.760	
200	4	1	1		0.767	0.756	
200	8	1	1		0.741	0.729	
200	16	0.990	0.990		0.760	0.745	
200	32	0.921	0.921		0.731	0.713	
200	64	0.821	0.820		0.690	0.667	
200	128	0.741	0.736		0.653	0.629	
400	2	1	1		0.764	0.753	
400	4	1	1		0.772	0.760	
400	8	1	1		0.764	0.751	
400	16	0.992	0.992		0.752	0.738	
400	32	0.927	0.927		0.734	0.718	
400	64	0.830	0.829		0.703	0.682	
400	128	0.748	0.743		0.663	0.640	
800	2	1	1		0.767	0.754	
800	4	1	1		0.768	0.755	

Continuation of Table						
No. Trees	No. Samples	Training Accuracy	Training score	F1-score	Testing Accuracy	Testing F1-score
800	8	1	1		0.763	0.750
800	16	0.995	0.995		0.759	0.745
800	32	0.924	0.924		0.732	0.714
800	64	0.832	0.830		0.704	0.680
800	128	0.743	0.738		0.666	0.644
End of Table						

Table D.4: Results With 30 Audio Sec. Clips Using Mel-spectrogram Features

Number of Trees	Number of Min. Samples Split	Training Accuracy	Training score	F1-score	Testing Accuracy	Testing F1-score
100	2	1	1		0.757	0.745
100	4	1	1		0.762	0.749
100	8	1	1		0.753	0.741
100	16	0.990	0.990		0.744	0.728
100	32	0.928	0.928		0.714	0.694
100	64	0.823	0.822		0.692	0.670
100	128	0.738	0.733		0.660	0.639
200	2	1	1		0.762	0.750
200	4	1	1		0.762	0.749
200	8	1	1		0.756	0.744
200	16	0.992	0.992		0.759	0.743
200	32	0.930	0.930		0.733	0.715

Continuation of Table						
No. Trees	No. Samples	Training Accuracy	Training score	F1-score	Testing Accuracy	Testing score
200	64	0.827	0.826		0.692	0.669
200	128	0.735	0.731		0.654	0.635
400	2	1	1		0.777	0.764
400	4	1	1		0.762	0.750
400	8	1	1		0.758	0.745
400	16	0.993	0.993		0.758	0.743
400	32	0.924	0.924		0.727	0.708
400	64	0.829	0.828		0.693	0.670
400	128	0.749	0.744		0.654	0.632
800	2	1	1		0.764	0.752
800	4	1	1		0.764	0.751
800	8	1	1		0.760	0.746
800	16	0.993	0.993		0.761	0.746
800	32	0.926	0.927		0.736	0.718
800	64	0.830	0.828		0.698	0.676
800	128	0.741	0.738		0.649	0.629
End of Table						

## D.2 XGBoost Classifier Parameter Experiments

Table D.5: Results Using MFCC Features

Max Depth	Learning Rate	No. Estimators	Subsample	Booster	Training Accuracy	Training F1-score	Testing Accuracy	Testing F1-score
4	0.03	100	1	gbtree	0.871	0.870	0.673	0.652
5	0.03	100	1	gbtree	0.941	0.942	0.713	0.694
6	0.03	100	1	gbtree	0.979	0.979	0.724	0.707
7	0.03	100	1	gbtree	0.995	0.995	0.729	0.712
8	0.03	100	1	gbtree	1.000	1.000	0.722	0.706
9	0.03	100	1	gbtree	1.000	1.000	0.724	0.709
10	0.03	100	1	gbtree	1.000	1.000	0.734	0.719
11	0.03	100	1	gbtree	1.000	1.000	0.732	0.716
12	0.03	100	1	gbtree	1.000	1.000	0.719	0.703
13	0.03	100	1	gbtree	1.000	1.000	0.722	0.706
10	0	100	1	gbtree	0.176	0.050	0.144	0.042
10	0.001	100	1	gbtree	0.956	0.956	0.670	0.648
10	0.005	100	1	gbtree	0.992	0.992	0.700	0.681
10	0.01	100	1	gbtree	0.997	0.997	0.716	0.698
10	0.03	100	1	gbtree	1.000	1.000	0.734	0.719
10	0.05	100	1	gbtree	1.000	1.000	0.729	0.714
10	0.1	100	1	gbtree	1.000	1.000	0.744	0.732
10	0.5	100	1	gbtree	1.000	1.000	0.746	0.734
10	0.9	100	1	gbtree	1.000	1.000	0.733	0.720
10	1	100	1	gbtree	1.000	1.000	0.737	0.723
10	0.5	50	1	gbtree	1.000	1.000	0.747	0.735
10	0.5	100	1	gbtree	1.000	1.000	0.746	0.734

Continuation of Table									
Max Depth	Learning Rate	No. Estimators	Subsample	Booster	Training Accuracy	Training F1-score	Testing Accuracy	Testing F1-score	
10	0.5	125	1	gbtree	1.000	1.000	0.746	0.734	
10	0.5	150	1	gbtree	1.000	1.000	0.748	0.735	
10	0.5	200	1	gbtree	1.000	1.000	0.752	0.740	
10	0.5	400	1	gbtree	1.000	1.000	0.747	0.734	
10	0.5	500	1	gbtree	1.000	1.000	0.749	0.736	
10	0.5	600	1	gbtree	1.000	1.000	0.748	0.735	
10	0.5	800	1	gbtree	1.000	1.000	0.748	0.735	
10	0.5	1000	1	gbtree	1.000	1.000	0.747	0.734	
10	0.5	1200	1	gbtree	1.000	1.000	0.748	0.735	
10	0.5	1500	1	gbtree	1.000	1.000	0.746	0.733	
10	0.5	200	1	gbtree	1.000	1.000	0.752	0.740	
10	0.5	200	0.9	gbtree	1.000	1.000	0.742	0.729	
10	0.5	200	0.8	gbtree	1.000	1.000	0.742	0.729	
10	0.5	200	0.7	gbtree	1.000	1.000	0.746	0.734	
10	0.5	200	0.6	gbtree	1.000	1.000	0.742	0.731	
10	0.5	200	0.5	gbtree	1.000	1.000	0.737	0.725	
10	0.5	200	0.4	gbtree	1.000	1.000	0.741	0.729	
10	0.5	200	0.3	gbtree	1.000	1.000	0.744	0.732	
10	0.5	200	0.2	gbtree	1.000	1.000	0.724	0.711	
10	0.5	200	0.9	gbtree	0.989	0.989	0.713	0.697	
10	0.5	200	1	gbtree	1.000	1.000	0.752	0.740	
10	0.5	200	1	gblinear	0.581	0.577	0.562	0.550	

Continuation of Table								
Max Depth	Learning Rate	No. Estimators	Subsample	Booster	Training Accuracy	Training F1-score	Testing Accuracy	Testing F1-score
10	0.5	200	1	dart	1.000	1.000	0.752	0.740
End of Table								

Table D.6: Results Using Mel-Spectrogram Features

Max Depth	Learning Rate	No. Estimators	Subsample	Booster	Training Accuracy	Training F1-score	Testing Accuracy	Testing F1-score
4	0.03	100	1	gbtree	0.882	0.882	0.694	0.677
5	0.03	100	1	gbtree	0.960	0.8960	0.722	0.705
6	0.03	100	1	gbtree	0.994	0.994	0.744	0.728
7	0.03	100	1	gbtree	1.000	1.000	0.741	0.726
8	0.03	100	1	gbtree	1.000	1.000	0.711	0.694
9	0.03	100	1	gbtree	1.000	1.000	0.719	0.704
10	0.03	100	1	gbtree	1.000	1.000	0.728	0.713
11	0.03	100	1	gbtree	1.000	1.000	0.720	0.705
12	0.03	100	1	gbtree	1.000	1.000	0.727	0.711
13	0.03	100	1	gbtree	1.000	1.000	0.728	0.713
7	0	100	1	gbtree	0.176	0.050	0.144	0.042
7	0.001	100	1	gbtree	0.891	0.891	0.646	0.629
7	0.005	100	1	gbtree	0.949	0.950	0.692	0.674
7	0.01	100	1	gbtree	0.975	0.975	0.698	0.678
7	0.03	100	1	gbtree	1.000	1.000	0.741	0.726
7	0.05	100	1	gbtree	1.000	1.000	0.750	0.737

Continuation of Table									
Max Depth	Learning Rate	No. Estimators	Subsample	Booster	Training Accuracy	Training F1-score	Testing Accuracy	Testing F1-score	
7	0.1	100	1	gbtree	1.000	1.000	0.752	0.739	
7	0.5	100	1	gbtree	1.000	1.000	0.746	0.735	
7	0.9	100	1	gbtree	1.000	1.000	0.742	0.733	
7	1	100	1	gbtree	1.000	1.000	0.742	0.733	
7	0.1	50	1	gbtree	1.000	1.000	0.740	0.726	
7	0.1	100	1	gbtree	1.000	1.000	0.752	0.739	
7	0.1	125	1	gbtree	1.000	1.000	0.753	0.742	
7	0.1	150	1	gbtree	1.000	1.000	0.754	0.743	
7	0.1	200	1	gbtree	1.000	1.000	0.758	0.747	
7	0.1	400	1	gbtree	1.000	1.000	0.762	0.751	
7	0.1	500	1	gbtree	1.000	1.000	0.754	0.744	
7	0.1	600	1	gbtree	1.000	1.000	0.756	0.745	
7	0.1	800	1	gbtree	1.000	1.000	0.758	0.747	
7	0.1	1000	1	gbtree	1.000	1.000	0.757	0.746	
7	0.1	1200	1	gbtree	1.000	1.000	0.751	0.740	
7	0.1	1500	1	gbtree	1.000	1.000	0.752	0.742	
7	0.1	400	1	gbtree	1.000	1.000	0.762	0.751	
7	0.1	400	0.9	gbtree	1.000	1.000	0.751	0.739	
7	0.1	400	0.8	gbtree	1.000	1.000	0.744	0.733	
7	0.1	400	0.7	gbtree	1.000	1.000	0.748	0.737	
7	0.1	400	0.6	gbtree	1.000	1.000	0.753	0.743	
7	0.1	400	0.5	gbtree	1.000	1.000	0.751	0.741	



Continuation of Table								
Max Depth	Learning Rate	No. Estimators	Subsample	Booster	Training Accuracy	Training F1-score	Testing Accuracy	Testing F1-score
7	0.1	400	0.4	gbtree	1.000	1.000	0.766	0.756
7	0.1	400	0.3	gbtree	1.000	1.000	0.750	0.739
7	0.1	400	0.2	gbtree	1.000	1.000	0.746	0.736
7	0.1	400	0.1	gbtree	0.994	0.994	0.748	0.739
7	0.1	400	0.4	gbtree	1.000	1.000	0.766	0.756
7	0.1	400	0.4	gblinear	0.660	0.660	0.611	0.596
7	0.1	400	0.4	dart	1.000	1.000	0.747	0.735
End of Table								

D.3 Multi Layer Perception Classifier

Table D.7: Multi Layer Perception Classifier Results on Different Activation Using MFCC Features

	ReLU	Identity	Logistic	tanH
Test Accuracy	0.469	0.457	0.732	0.663
Test F-score	0.381	0.441	0.716	0.645
Test Precision	0.444	0.552	0.723	0.645
Test Recall	0.457	0.449	0.729	0.656
End of Table				

Table D.8: Multi Layer Perception Classifier Results on Different Solver Using MFCC Features

	<b>Adam</b>	<b>LBFGS</b>	<b>SGD</b>
Test Accuracy	0.719	0.561	0.160
Test F-score	0.706	0.549	0.046
Test Precision	0.702	0.549	0.027
Test Recall	0.712	0.558	0.167
End of Table			

Table D.9: Multi Layer Perception Classifier Results on Different Learning Rate Schedules Using MFCC Features

	<b>Constant</b>	<b>invscaling</b>	<b>adaptive</b>
Test Accuracy	0.742	0.723	0.706
Test F-score	0.725	0.707	0.687
Test Precision	0.729	0.711	0.697
Test Recall	0.739	0.719	0.701
End of Table			

Table D.10: Multi Layer Perception Classifier Results on Different Initial Learning Rates Using MFCC Features

<b>Learning Rate</b>	<b>Test Acc.</b>	<b>Test F-score</b>	<b>Test Precision</b>	<b>Test Recall</b>
0.005	0.722	0.708	0.706	0.717
0.003	0.754	0.740	0.739	0.751
0.001	0.749	0.735	0.732	0.745
0.0008	0.761	0.751	0.752	0.756
0.0006	0.737	0.723	0.724	0.729
0.0004	0.746	0.732	0.732	0.742

Continuation of Table				
Learning Rate	Test Acc.	Test F-score	Test Precision	Test Recall
0.0002	0.746	0.733	0.733	0.740
0.0001	0.684	0.671	0.671	0.678
End of Table				

Table D.11: Multi Layer Perception Classifier Results on Different Activation Using Mel-spectrogram Features

	ReLU	Identity	Logistic	tanH
Test Accuracy	0.626	0.460	0.138	0.103
Test F-score	0.609	0.456	0.098	0.102
Test Precision	0.613	0.471	0.195	0.111
Test Recall	0.622	0.456	0.136	0.102
End of Table				

Table D.12: Multi Layer Perception Classifier Results on Different Solver Using Mel-spectrogram Features

	Adam	LBFGS	SGD
Test Accuracy	0.668	0.163	0.144
Test F-score	0.653	0.102	0.042
Test Precision	0.650	0.196	0.024
Test Recall	0.663	0.167	0.167
End of Table			

Table D.13: Multi Layer Perception Classifier Results on Different Learning Rate Schedules Using Mel-spectrogram Features

	Constant	invscaling	adaptive
Test Accuracy	0.696	0.649	0.702
Test F-score	0.684	0.626	0.686
Test Precision	0.685	0.626	0.681
Test Recall	0.691	0.642	0.696
End of Table			

Table D.14: Multi Layer Perception Classifier Results on Different Initial Learning Rates Using Mel-spectrogram Features

Learning Rate	Test Acc.	Test F-score	Test Precision	Test Recall
0.005	0.640	0.627	0.634	0.634
0.003	0.654	0.637	0.657	0.647
0.001	0.706	0.694	0.694	0.700
0.0008	0.674	0.641	0.644	0.669
0.0006	0.657	0.643	0.652	0.651
0.0004	0.673	0.654	0.649	0.668
0.0002	0.681	0.662	0.656	0.675
0.0001	0.672	0.652	0.646	0.666
End of Table				

## D.4 Support Vector Machines

Table D.15: Support Vector Machine Performance Results on Different Kernel Types using MFCC Features

	<b>Linear</b>	<b>Poly</b>	<b>Sigmoid</b>	<b>RBF</b>
Test Accuracy	0.327	0.283	0.122	0.289
Test F-score	0.277	0.233	0.048	0.227
Test Precision	0.552	0.259	0.057	0.223
Test Recall	0.328	0.285	0.140	0.291
End of Table				

Table D.16: Support Vector Machine Performance Results on Different Kernel Types using Mel-spectrogram Features

	<b>Linear</b>	<b>Poly</b>	<b>Sigmoid</b>	<b>RBF</b>
Test Accuracy	0.450	0.430	0.337	0.571
Test F-score	0.406	0.444	0.327	0.566
Test Precision	0.438	0.633	0.342	0.577
Test Recall	0.433	0.423	0.333	0.565
End of Table				

## D.5 Logistic Regression

Table D.17: Logistic Regression Performance Results on Different Solvers using MFCC Features

	<b>LBFGS</b>	<b>SAG</b>	<b>SAGA</b>	<b>Newton-CG</b>
Test Accuracy	0.540	0.546	0.542	0.558
Test F-score	0.521	0.520	0.512	0.545
Test Precision	0.514	0.515	0.511	0.539

Continuation of Table				
	<b>LBFGS</b>	<b>SAG</b>	<b>SAGA</b>	<b>Newton-CG</b>
Test Recall	0.537	0.541	0.538	0.556
End of Table				

Table D.18: Logistic Regression Performance Results on Different Solvers using Mel-spectrogram Features

	<b>LBFGS</b>	<b>SAG</b>	<b>SAGA</b>	<b>Newton-CG</b>
Test Accuracy	0.579	0.521	0.517	0.602
Test F-score	0.563	0.508	0.504	0.584
Test Precision	0.559	0.504	0.500	0.579
Test Recall	0.573	0.515	0.511	0.594
End of Table				

## Appendix E

### Comparison of Deep Learning Models in Present Literature With This Study

## E.1 TDNN Architecture

Table E.1: Performance of Models in Present Studies Using TDNN

Model	Research Title	Input Feature	Emotion Labels	Dataset	Dataset Category	Validation Accuracy Score
TDNN Dense N-SER	Using Convolutional Neural Networks for Naturalistic Speech Emotion Recognition	MFCC 40-dimensional	Happy, Sad, Angry, Surprise, Fear, Neutral	IEMOCAP	Spontaneous	61.44%
TDNN Convolutional N-SER		Mel-spectrogram 128 Mel-scales				61.22%
		MFCC 40-dimensional				70.22%



Continuation of Table						
Model	Research Title	Input Feature	Emotion Labels	Dataset	Dataset Category	Validation Accuracy Score
		Mel-spectrogram 128 Mel-scales				62.56%
5-fold TDNN with Transfer Learning	A Transfer Learning Method for Speech Emotion Recognition from Automatic Speech Recognition	MFCC with iVector-based features	Angry, Excited, Sad, Neutral	IEMOCAP	Induced	71.7%

Continuation of Table						
Model	Research Title	Input Feature	Emotion Labels	Dataset	Dataset Category	Validation Accuracy Score
X-vector TDNN Architecture  ECAPA-TDNN Architecture	Applying TDNN Architectures for Analyzing Duration Dependencies on Speech Emotion Recognition	MFCC 40-dimensional	Angry, Happy, Sad, Neutral	IEMOCAP	Induced	50.71%
	Emotion Identification from Raw Speech Signals Using DNNs	MFCC 23-dimensional	Angry, Happy, Sad, Neutral	IEMOCAP	Induced	58.67%
TDNN with Statistics Pooling						55.30%

Continuation of Table					
Model	Research Title	Input Feature	Emotion Labels	Dataset	Dataset Category
End of Table					
					Validation Accuracy Score

## E.2 ResNet50 Architecture

Table E.2: Performance of Models in Present Studies Using ResNet50

Model	Research Title	Input Feature	Emotion Labels	Dataset	Dataset Category	Validation Accuracy Score
ResNet50 N-SER	Using Convolutional Neural Networks for Naturalistic Speech Emotion Recognition (N-SER)	MFCC 40-dimensional	Happy, Sad, Angry, Surprise, Fear, Neutral	IEMOCAP	Spontaneous	65.44%

Continuation of Table						
Model	Research Title	Input Feature	Emotion Labels	Dataset	Dataset Category	Validation Accuracy Score
		Mel-spectrogram 128 Mel-scales				49.89%
ResNet50	LSSED: A Large-Scale Dataset and Benchmark for Speech Emotion Recognition	Mel-spectrogram 128 Mel-scales	Angry, Happy, Sad, Neutral	LSSED	Naturalistic	37.70%

Continuation of Table						
Model	Research Title	Input Feature	Emotion Labels	Dataset	Dataset Category	Validation Accuracy Score
ResNet50	Deep Neural Network for Visual Emotion Recognition Based on ResNet50 Using Song-Speech Characteristics	Facial Features	Angry, Calm, Fear, Happy, Neutral, Sad	RAVDESS	Posed	55.52%
End of Table						

### E.3 CNN+LSTM Architecture

Table E.3: Performance of Models in Present Studies Using CNN+LSTM

Model	Research Title	Input Feature	Emotion Labels	Dataset	Dataset Category	Validation Accuracy Score
CNN + N-SER	Using Convolutional Neural Networks for Speech Emotion Recognition (N-SER)	MFCC 40-dimensional  Mel-spectrogram 128 Mel-scales	Happy, Sad, Angry, Surprise, Fear, Neutral	IEMOCAP	Spontaneous	63.89%
1D CNN + LSTM	Speech Emotion Recognition Using Deep 1D & 2D CNN LSTM Networks	Mel-spectrogram	Angry, Excited, Frustrated, Happy, Neutral, Sad	IEMOCAP	Induced	62.07%
End of Table						

## E.4 CNN Architecture

Table E.4: Performance of Models in Present Studies Using CNN

Model	Research Title	Input Feature	Emotion Labels	Dataset	Dataset Category	Validation Accuracy Score
Base CNN N-SER	Using Convolutional Neural Networks for Naturalistic Speech Emotion Recognition (N-SER)	MFCC 40-dimensional	Happy, Sad, Angry, Surprise, Fear, Neutral	IEMOCAP	Spontaneous	63.89%
		Mel-spectrogram 128 Mel-scales				64.00%

Continuation of Table						
Model	Research Title	Input Feature	Emotion Labels	Dataset	Dataset Category	Validation Accuracy Score
1D CNN + LSTM	Speech Emotion Recognition Using Deep 1D & 2D CNN LSTM Networks	Mel-spectrogram	Angry, Excited, Frustrated, Happy, Neutral, Sad	IEMOCAP	Induced	62.07%
CNN	Speech Emotion Recognition Using Convolutional Neural Networks	Log Mel-spectrogram	Angry, Happy, Sad, Neutral	IEMOCAP	Induced	59.33%



Continuation of Table						
Model	Research Title	Input Feature	Emotion Labels	Dataset	Dataset Category	Validation Accuracy Score
CNN	A Comprehensive Study On Bilingual and Multilingual Speech Emotion Recognition Using a Two-Pass Classification Scheme	MFCC 12-dimensional	Angry, Happy, Sad, Neutral	IEMOCAP	Induced	55.20%
End of Table						

# Appendix F

## Resource Persons

**Dr. Merlin Teodosia Suarez**

Adviser

College of Computer Studies

De La Salle University-Manila

`merlin.suarez@dlsu.edu.ph`

**Mr. Chuan-chen Chu**

Student

College of Computer Studies

De La Salle University-Manila

`chuan-chen.chu@dlsu.edu.ph`

**Mr. Reynaldo Delima Jr.**

Student

College of Computer Studies

De La Salle University-Manila

`reynaldo_delimajr@dlsu.edu.ph`

**Mr. Nilo Cantil Jatico II**

Student

College of Computer Studies

De La Salle University-Manila

`nilo_jaticooii@dlsu.edu.ph`

**Mr. Jedwig Siegfrid Tan**

Student

College of Computer Studies

De La Salle University-Manila

`jedwig_siegfrid.tan@dlsu.edu.ph`

