Emotion Recognition through voice by MLP classifier and Deep Sequential Neural Network

by

Armina Salemi

Bachelor, Iran University of Science and Technology, 2015

Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Applied Science

in the
School of Mechatronic Systems Engineering
Faculty of Applied Sciences

SIMON FRASER UNIVERSITY
Fall 2021

Declaration of Committee

Name:	**Armina Salemi**
Degree:	**Master of Applied Science**
Title:	**Emotion Recognition through voice by MLP classifier and Deep Sequential Neural Network**
Committee:	**Chair:** **Mehrdad Moallem** Professor, Mechatronic Systems Engineering
	Ahmad Rad Supervisor Professor, Mechatronic Systems Engineering
	Mohammad Narimani Committee Member Lecturer, Mechatronic Systems Engineering
	Shahram Payandeh Examiner Professor, Engineering Science

Abstract

This thesis aspires to provide a thorough study on Speech Emotion Recognition in the field of Machine Learning. The main objective is to simplify the path towards emotion recognition by voice without sacrificing efficiency. In other words, the presented thesis studies the limits of least computationally complex algorithms in SER. By the end of this study, a traditional method which is MLP classifier and a Deep Learning approach with Deep Sequential Neural Network are compared. The algorithms use RAVDESS for training. Different combinations of three features, MFCC, Mel-spectrogram, and Chroma are utilized in both algorithms to determine the most efficient combination.

Keywords: Emotion Recognition; Deep Learning; MLP classifier; Voice Signal; Speech Emotion Recognition

Dedication

This thesis is wholeheartedly dedicated to my parents for ever supporting me, and my partner for always believing in me. I would never have made it this far without your love and patience.

Acknowledgements

First and foremost, I would like to express my gratitude towards my supervisor, Dr Ahmad Rad for showing me a whole new world and advising me all the way through it. This project is completely inspired by you, and your encouragement shaped a dream into reality. I offer my sincere appreciation for the opportunities you provided.

Second, I appreciate my lab-mates, Mohammad, Kamal, Elfituri, and Mehdi for making me feel welcomed and helping me in my studies with their experience and patience. Also, my never-ending gratitude to my friend Parinaz for helping me overcoming many difficulties in the process of getting accustomed to the new environment; and my roommate Pegah, for her adventurous spirit and easy-going personality that made everything easier.

In the end, my deepest appreciation to my most beloved ones: my family. It was not an easy path and I have been going through many hardships. I could have not fought this hard if you were not lending me your strength. Thank you.

Table of Contents

List of Tables

List of Figures

Chapter 1.

Introduction

1.1. Background

Among human senses, sight and hearing are central in interactions with the outside world. We inadvertently respond to sounds around us; yet conscientious hearing (listening) has contributed to understanding of intent; consequently, voice is regarded as the principal medium for articulation of thoughts, ideas, and emotions. Whereas thoughts and ideas are communicated seamlessly through voice, sight is a far superior modality for detection of emotions. Both sight and hearing modalities can detect extreme emotions (happiness or anger) effortlessly; however, vision is more adept to detect subtle emotions. It is well argued in literature that detection of emotions is a multi-modal problem as all senses are collectively at work.

With the advent of social robots that could be deployed in every home [1] in the next two decades, a significant research interest is devoted to emotion detection by machines. As the popularity of social robots and virtual assistants grows, Artificial Intelligence community is more focused on methods to improve Human-Computer Interaction (HCI). To achieve the objective of "humanizing" interactions of man and machine, Affective Computing (AC), first introduced by Rosalind Picard in 1997 [2], appears to be a viable solution. In her seminal paper, she defined AC as: *"computing that relates to, arises from, or influences emotions."* [3]. Moreover, she predicted that not only expressing and recognizing emotions, but computers soon would be able to "*have emotions*". She also surmised that humans make their decisions with both thoughts and emotions; therefore, in order to pass the Turing Test, a computer must acquire the ability of affect recognition at some basic levels (at the very least).

Tao and Tan [4] elaborated and summarized the different approaches regarding affective computing, dividing them into five sections: Emotional speech processing, facial expression, body gesture and movement, multimodal systems and affect understanding and cognition. Poria [5], however, narrowed them down into two divisions: Audial data and Visual data as shown in Figure 1.1.

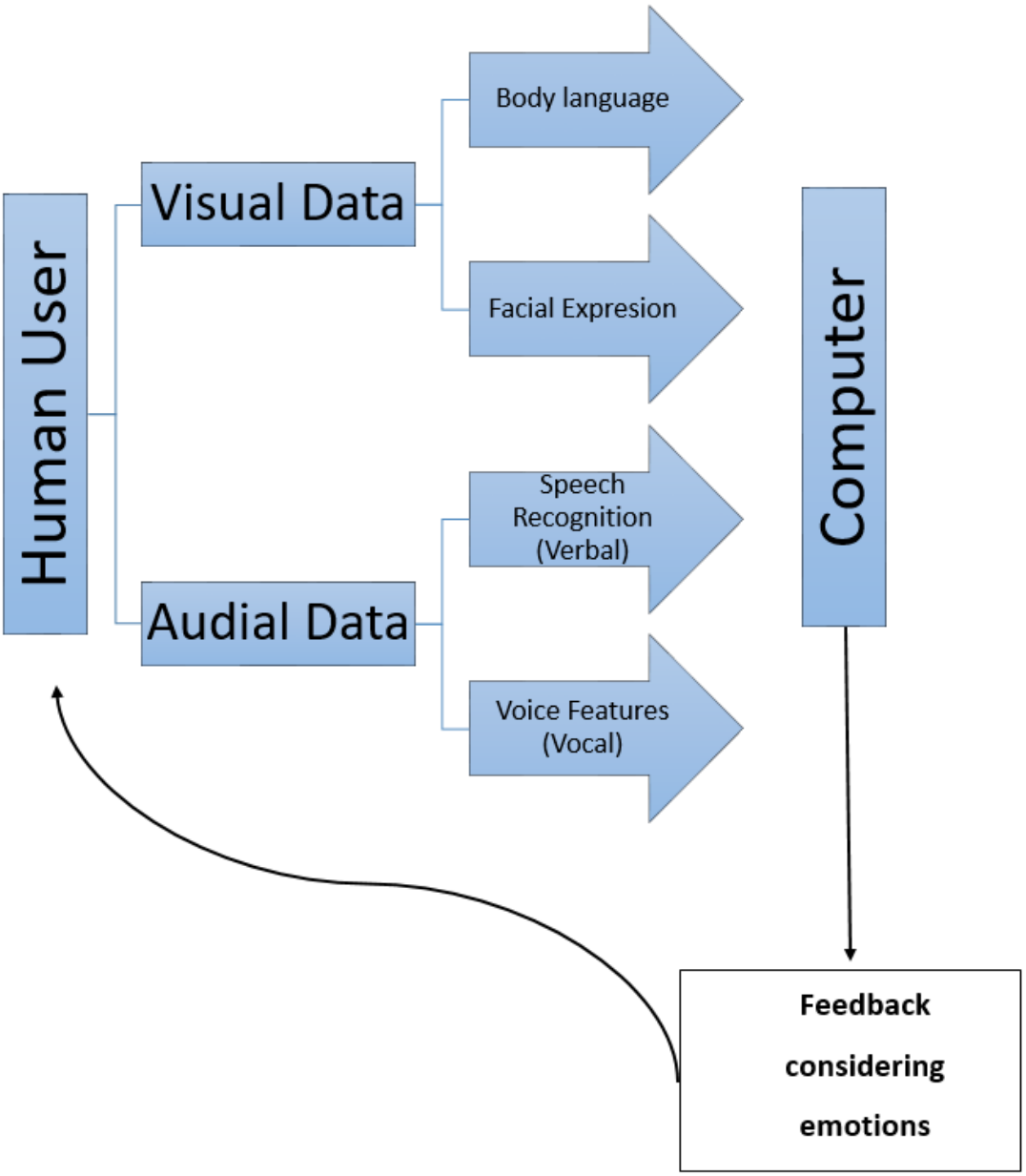

Figure 1.1. A typical multimodal affect analysis framework

In summary, for a natural HCI, it is not enough for a robot to understand its user's intents via conventional inputs. The current technology allows HCI through voice and vision. Moreover, in many scenarios, vision is not an option (e.g., Call Centers, Customer Service Lines, etc.); in such instances, Speech Emotion Recognition (SER) permits seamless communication. As the systems move from the laboratory settings to real world situations, major advancements are required to make them efficient in

unsupervised situations. The main objective of the studies reported in this thesis is to study emotion detection algorithms that can be implemented in real systems (like humanoid robots) with modest computation. Another objective of this thesis is to provide comparative simulation studies among selected algorithms and discuss their merits for real-time implementations.

Before anything, in order to design and develop a successful algorithm, not only the understanding of coding and different methods needed is requisite, but also one requires the knowledge of the underlying theory and principal models for the purpose in hand. Therefore, the background section of this chapter starts with a brief review on Emotion Models. The section continues with Traditional Methods for Emotion Recognition through voice and ends after elaborating the need and algorithms for Deep Learning methods.

1.1.2 Emotion Models

Klaus R. Scherer [6] contends that the majority of accepted theories come up with “multi-componential definition”. However, there is another school of thought led by Sinclair et al. [7] that advocates against using the term “emotion” to a single modality as it would restrict it to conscious feelings of changed states. Further studies elaborate the fact that there are several convergences in this area and discussions are on-going suggesting that there are considerable differences among scholars on Emotion Models. Moreover, it is suggested that emotions depend on cultural and social characteristics. Even among the theorists who believe emotions in general or at least some of them are innate, there are indications towards two subclasses of “basic” and “fundamental” emotions [8]. There are major disagreements as to which emotions are they, though.

Having that said, most of theorists categorize emotions by these two labels: Primary and Secondary. Considering the neuropsychology aspect, it is believed that primary emotions are innate and would be initiated “in the lower-level limbic system of the brain” [9]. In other words, emotions such as anger or fear may take over the body responses before any signals causes cortex starts analyzing an event. On the other hand, secondary emotions like acceptance or annoyance are the next level when brain had initiated reasoning.

Scherer believes all of Emotion Models may fall in either one of these classes: Dimensional (or Appraisal Theory), Discrete, Meaning Oriented and Componential Models [6]. For a simpler approach, the models could be categorized into two classes: Categorical which presents exact and discrete categories and Dimensional which consider several dimensions (e.g., Intensity, Valence, etc.) and specify each emotion by its coordination. Having them all mentioned, a new affective model referred to “The Hourglass of Emotions” was presented in 2012 by the name of “The Hourglass of Emotions” which clusters emotions both categorically and dimensionally [10]. Note that all the models regardless of their type are based still believe in the episodic nature of emotion.

1.1.3 Traditional Approaches towards SER

The initial efforts for Speech Emotion Recognition tasks, which are considered traditional methods, consist of two main stages: Feature extraction and classification. The features may vary regarding the application, as in some models, speech rate or silence duration is considered. However, they do not necessarily depend on the speaker or lexical information. Having that mentioned, there are times when the aural information on HOW something is said does not suffice. In that case, linguistic analysis is another option to clarify WHAT is said. Expectedly, this combination (Voice analysis and Speech analysis) leads to a better result.

However, before the two steps of feature extraction and categorization, in order to preprocess the data, there are signal processing, de-noising, and segmentation to deliver meaningful units of the file for classification. Afterwards, there is feature extraction stage in which relevant characteristics of data is extracted and determined how they are associated with each class of emotions. After mapping the features to under study categories, the classification task would be carried out. A schematic of the whole procedure is presented in Figure 1.2.

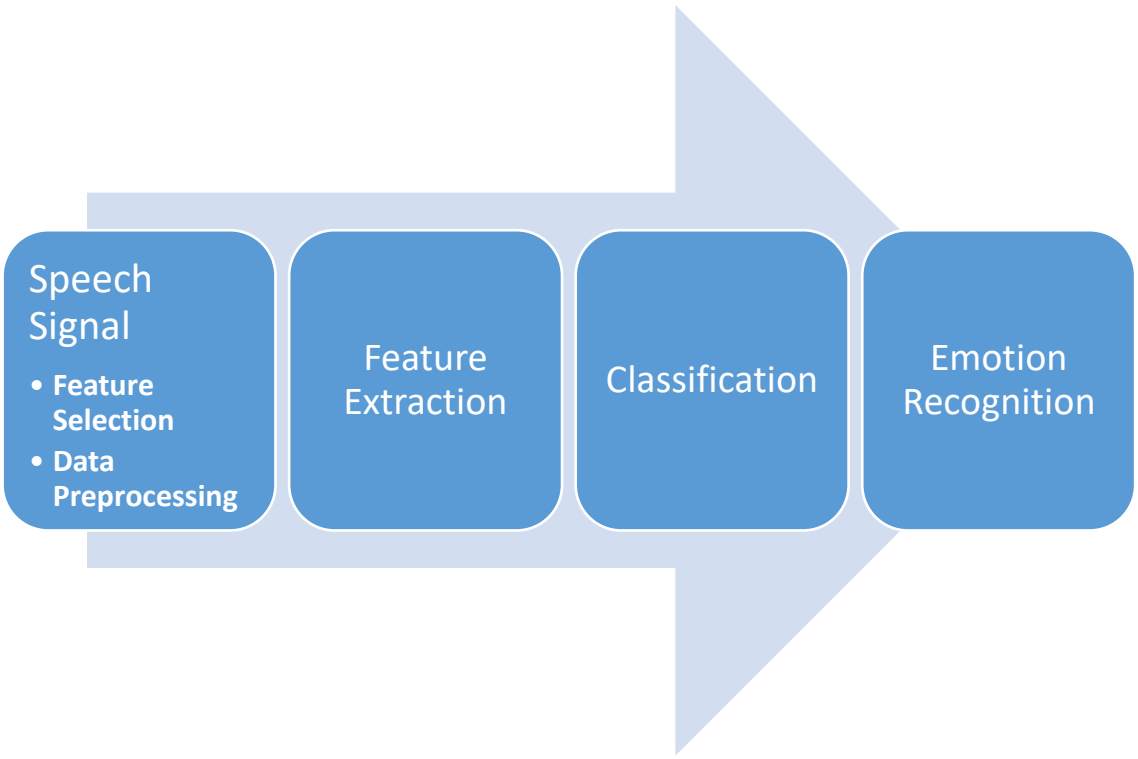

Figure 1.2. The Process of Traditional Emotion Recognition

There are various linear or non-linear approaches toward Speech Emotion Recognition. Support Vector Machine (SVM) is among the most popular linear models and for non-linear models, Gaussian Mixture Models (GMMs) [11] and Hidden Markov Models (HMMs) are the most known classifiers. An example for GMM is [12] and examples for HMM classifier are [13] and [14] which the latter HMM classifies specifically through MFCC features. More examples and detailed background are discussed in Chapter 2 (literature review) of the thesis.

1.1.4. Deep Learning Methods for SER

During the era of Deep Learning methods, it is not an exception to employ them in Speech Emotion Recognition but a popular trend. In the beginning, the main disadvantage of Deep Learning methods used to be the fact that they require considerable amount of data for their training. However, the Internet and online data have come to the rescue, providing immense training data. Hence, researchers show more and more interest in Deep Learning methods for Speech Emotion Recognition tasks. Moreover, Deep Learning methods integrate feature extraction and classification within the network. Nonetheless, as they offer this option in addition to the higher

accuracies they usually achieve in comparison to traditional methods, they have become very popular in recent years. In spite of all aforementioned advantages, the training stage requires extensive computation and significantly higher processing power.

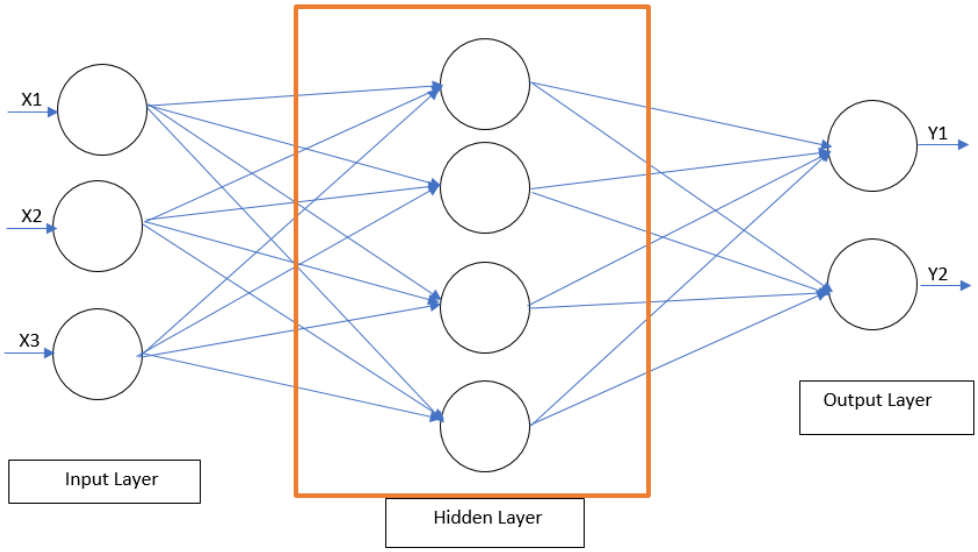

Figure 1.3. Feedforward Neural Network (Can be considered Deep if Hidden Layers>1)

A Deep Learning procedure can be carried out through three different approaches: Supervised, Unsupervised, and Semi-supervised. It necessarily possesses more than three layers, including input and output layers which implies more than one hidden layer. Bengio provides keen readers with more detailed information regarding Deep Learning algorithms generally for any AI task [15] (e.g., Natural Language Processing, Speech and Image Recognition, etc.) while Khalil et al. is focused on Speech Emotion Recognition in details [16]. Additionally, Figure 1.3 presents a schematic of a Feedforward Neural Network alongside the primary condition that changes them into a Deep Neural Network in the caption below it.

In Speech Emotion Recognition field, there are various options among Deep Learning approaches. A popular example is Convolutional Neural Networks which works by visualizing the speech signal and recognising the pattern of each emotion in an image. Recurrent Neural Networks are another option that are widely used in Natural Language Processing field due to their method for voice signals framing in short-time levels [17].

1.2. Motivation

The advancement of Artificial Intelligence in general leads to the extensive interest in social robots. In recent years, researchers and programmers are dedicated to improving social robots as human companion and one of the approaches is through affective computing. One of main differences between a human companion and an artificial one is that the latter used to be incapable of perceiving and understanding the emotion, hence, unable to modify its reaction respectively; that is why Affective Computing combined with social robots such as NAO robots has become an emerging field.

However, even though there are varieties of algorithm for performing the task, it is not yet obvious what the best criteria are for Speech Emotion Recognition. Moreover, due to progresses in hardware field, specifically processors, algorithms tend to become more complex, while there is another option: augmenting straight-forward algorithms with suitable features, generating accurate models that do not require heavy processing. The main question is whether or not it is possible to implement an algorithm, not necessarily deep, on a social robot or virtual assistant without sacrificing accuracy and time. Studies have been conducted to compare complex Deep Neural Network with each other and determine the most accurate, while there is not many experiments and comparison between traditional and simple deep networks.

The first objective of this thesis is to observe the difference in accuracy between a Deep Sequential Neural Network and a Multi-Layer Perceptron Classifier. The second objective is to decide on the features that improve each, if any. The impact of the three features used for Emotion Recognition by voice signals is also studied throughout this thesis.

1.3. Thesis structure

The overview of this thesis and the structure are as follows.

Chapter 2 contains a selected literature review on the field of Emotion Recognition. The survey starts with an extensive look over emotion models, follows with different approaches for emotion recognition in general. After that there is a review on

available datasets and their types and languages. The chapter also includes a section for feature extraction and classifications, which are related to traditional methods. Deep Learning and its algorithms are reviewed in the following section of this chapter. The chapter concludes with future directions and conclusion sections.

Chapter 3 includes a comparative analysis of performance of three feature extraction algorithms applied to multi-layer perceptron classifier. It starts with an introduction that is followed by an observation on features which are utilized in this study. Before conclusion, the chapter goes through the algorithm and its confusion matrices.

Chapter 4 is focused on the Deep Learning method, however, as the dataset and features are the same for both algorithms, the details regarding them are dismissed in this chapter. Chapter four contains an introduction, approach, and results after removing each feature to study their affects. Attaching confusion matrices for each step, the chapter ends with the conclusion in what is the best combination of features and what are the drawbacks regarding detecting seven primary emotion and neutral state.

Chapter 5 is Comparison and Conclusion. At first the two algorithms and their results are compared under different circumstances. “Different circumstances” means a feature is removed in each step from algorithms and the accuracies are compared. In the process, the impact of the feature is studies. After the conclusion, future directions of research in this area are outlined.

Chapter 2.

Literature review

2.1. Introduction

As virtual assistants like Siri [16] and Alexa [17] and social robots such as NAO robots [18] become more popular, Human-Computer Interaction (HCI) via voice interface has attracted significant attention from robotics researchers as well as software developers as a natural way of communication. This is where Affective Computing merges with AI. This area has attracted the attention of many researchers in the last few years. One should not underestimate the complexity of the task as recognition of emotions by voice alone proves to be difficult for humans as well as machines.

There are several parameters including but not limited to cultural differences, genders, personality, etc. that could affect how humans express their emotions; as such, the concept of emotion itself is a complicated subject. There are many instances whereby both sides of the conversation are not physically present; hence, one does not have the access to all kind of information from visual data (e.g., body language, facial expression, etc.) to verbal and vocal inputs. This is why researchers have addressed emotion detection from different modalities. Some studies provide observations on Emotion Recognition through text that is helpful in cases like analysing online reviews and feedbacks for websites [19]. There are other studies that consider both Speech Emotion Recognition (SER) and emotion detection from text [20]. Machine Learning (ML) techniques have been very popular to address emotion detection [20]. Moreover, with the advent of Deep Learning, a significant body of research has been generated towards SER [21] which concentrates on Deep Learning approaches towards SER and provides a review on them. D'mello et al. provides a comparison of 64 different models up to 2015 [22]. The focus of that paper is on multimodal approaches and tries to cluster them by various labels such as Data Type or Modalities. Another review on both ML and Deep Learning techniques for Emotion Recognition by voice is reported in [23]. There are also reviews on techniques for emotion recognition through visual data [24]. However, since the focus of this thesis is on emotion recognition by speech signals, other approaches are neither critically examined nor covered in detail.

This field of study is mostly useful for call centers and costumer services (by phone). In these cases, there is a vast variety of information hidden in one's voice: pitches, speed, pauses, etc. Having that said, factors such as age, gender, language, and culture would affect all aforementioned parameters, which is why there is still work in progress for the task.

In this chapter, a selected review on most of the related topics to the main theme of the thesis is reported. In section 2.2, we present Models of emotions. In section 2.3, a brief discussion on emotion detection approaches are included. Datasets are discussed in section 2.4. A concise review of feature extraction and classification methods is reported in section 2.5. Deep learning is included in section 2.6. Towards the end of this literature review, a discussion on future direction for further developments is included in section 2.7. Finally, the chapter is concluded in section 2.8.

2.2. Models of Emotions

As mentioned before, emotion itself is a complicated topic and there is no complete agreement among psychologists on the most parts which leads to different emotion models and definitions. A brief review of those emotion models is provided in Table 2.1. Note that for the task of emotion recognition, due to its complexity, Categorical models with fewer classifications are the ones that are mostly investigated.

Table 2.1. Emotion Models

Model/Author	Years	Emotions	Approach
John Watson [25]	1928	Fear, love, rage	Categorical
Arnold [26]	1960	Anger, aversion, courage, dejection, desire, despair, fear, hate, hope, love, sadness	Categorical
Izard [27]	1971	Fear, anger, shame, contempt, disgust, guilt, distress, interest, surprise, joy	Categorical
Tomkins [28]	1984	Anger-rage, interest-excitement, distress-anguish, dismay-disgust, fear-terror, enjoyment-joy, shame-humiliation, surprise-startle	Categorical
Roseman [29]	1979/1984	Surprise, hope, joy, relief, fear, sadness, distress, frustration, disgust, liking, dislike, anger, contempt, pride, regret, guilt, shame	Componential

Plutchik [30]	1980	Acceptance, admiration, aggressiveness, amazement, anger, annoyance, anticipation, apprehension, awe, boredom, contempt, disapproval, disgust, distraction, ecstasy, fear, grief, interest, joy, loathing, love, optimism, pensiveness, rage, remorse, sadness, serenity, submission, surprise, terror, trust, vigilance	Dimensional
Circumplex Russell [31]	1980	Afraid, alarmed, angry, annoyed, aroused, astonished, at ease, bored, calm, content, delighted, depressed, distressed, droopy, excited, frustrated, glad, gloomy, happy, miserable, leased, relaxed, sad, satisfied, serene, sleepy, tense, tired	Dimensional
Ekman et al. [32]	1982	Anger, disgust, fear, joy, sadness, surprise	Categorical
The Component Process Model (CMP) [33]	1984	Satisfaction, joy, elation, triumph, hope, irritation, anger, fear, anxiety, sadness, rage	Componential
Weiner & Graham [34]	1984	Happiness, sadness	Categorical
Frijda et al. [35]	1986	Desire, happiness, interest, surprise, wonder, sorrow	Componential
Shaver et al. [36]	1987	Anger, fear, joy, love, sadness, surprise	Meaning Oriented
Oatley and Johnson-Laird [37]	1987	Anger, anxiety, disgust, happiness, sadness	Meaning Oriented
OCC/ Ortony et al. [38]	1988	Admiration, anger, appreciation, disappointment, disliking, fear, fears-confirmed, gloating, gratification, gratitude, happy-for, hope, liking, pity, pride, sorry-for, relief, remorse, reproach, resentment, self-reproach, shame	Dimensional
Lazarus-Smith [39]	1990	Anger, guilt, anxiety, sadness, hope	Dimensional
Lazarus [40]	1991a	Anger, fright, anxiety, guilt, shame, sadness, envy, jealousy, disgust, happiness, pride, relief, love, hope, compassion, gratitude	Componential
Lovheim [41]	2012	Anger/rage, contempt/disgust, distress/anguish, enjoyment/joy, fear/terror, interest/excitement, shame/humiliation, surprise/startle	Dimensional

Erik Cambria et al. [42]	2012	Rage, anger annoyance, apprehension, fear, terror Vigilance, anticipation, interest, distraction, surprise, amazement Ecstasy, joy, serenity, pensiveness, sadness, grief Admiration, trust, acceptance, boredom, disgust, loathing	Categorical and Dimensional

In Table 2.1, the models are categorized in four classes: Dimensional models which propose several dimensions (e.g., Intensity, Valence, etc.) and pinpoint each emotion by a coordination; Categorical models that believe in discrete nature of emotions and provides specific labels for each; Meaning oriented models which are prototypical mental representations and provides a description by words for each emotion; and Componential models which are a link between emotion evaluation and the elicited reaction patterns [36]. More details on these different categories are provided in [36] and are presented in Table 2.2.

Table 2.2. Details on classes of Emotion Models [36]

Models	Major Focus	Elicitation Mechanism	Differentiation Mechanism
Dimensional	Subjective feeling	Rarely directly addressed; rudimentary approach-avoidance definition	Degree of similarity on feeling dimension such as valence and activation
Discrete Emotion	Motor expression or adaptive behavior patterns	Rarely directly addressed; typical situation or stimulus configuration	Phylogenetically continuous neuroanatomical circuits or motor programs
Meaning	Verbal descriptions of subjective feelings	Rarely directly addressed; cultural interpretation patterns	Socially shared, prototypical mental representations
Componential	Link between emotion-antecedent evaluation and differentiated reaction patterns	Appraisal mechanism based on a universally valid set of criteria, influenced by cultural and individual differences	Adaptive reactions in motor expression; physiological responses to appraisal results and the action tendencies generated by the results

Even though there are many emotion models from which only a few is mentioned in Table 2.2, Affective Computing and Emotion Recognition based on Dimensional or Componential models are more complicated than discrete emotion models. Hence, most

approaches towards the task prefer specific labelling of Discrete Emotion Models and at most works on detecting seven primary emotions: Anger, calm, disgust, fear, happiness, sadness, surprise, alongside a Neutral state.

2.3. Emotion Detection approaches

Even if a simple emotion model is considered for detection and the labels include only primary emotions, emotion recognition is a difficult task.

Mehrabian [37] indicates there are three approaches towards communication: Verbal, Vocal, and Body Language which represent 7%, 38% and 55% of the communications, respectively, which is summarized in Figure 2.1. It is common sense that the main channel to convey a message is verbal. However, Hall suggests that the majority of received information regarding one's emotional state is not through language, and "occurs outside our awareness" [38]. This also implies that there are three different approaches and different methods to study a person's emotions. The approach might be through Verbal information (Speech Recognition: WHAT is said), Vocal Information (Speech Recognition: HOW it is said) and Visual Information considering one's body gesture or facial expression (Vision Recognition). Moreover, there are methods combining two or three of these approaches and lead to better detection; these approaches are referred to as Multimodal [31]. Also, it is possible to detect emotions from Bio-Potential Signals which requires headphones, EEG sensors, pulse meter, and other equipment. This method for recognition of five emotions (joy, anger, sadness, fear, and relax) achieved 41.7% success [39], though. The obvious problem with the latter approaches is the fact that they are invasive.

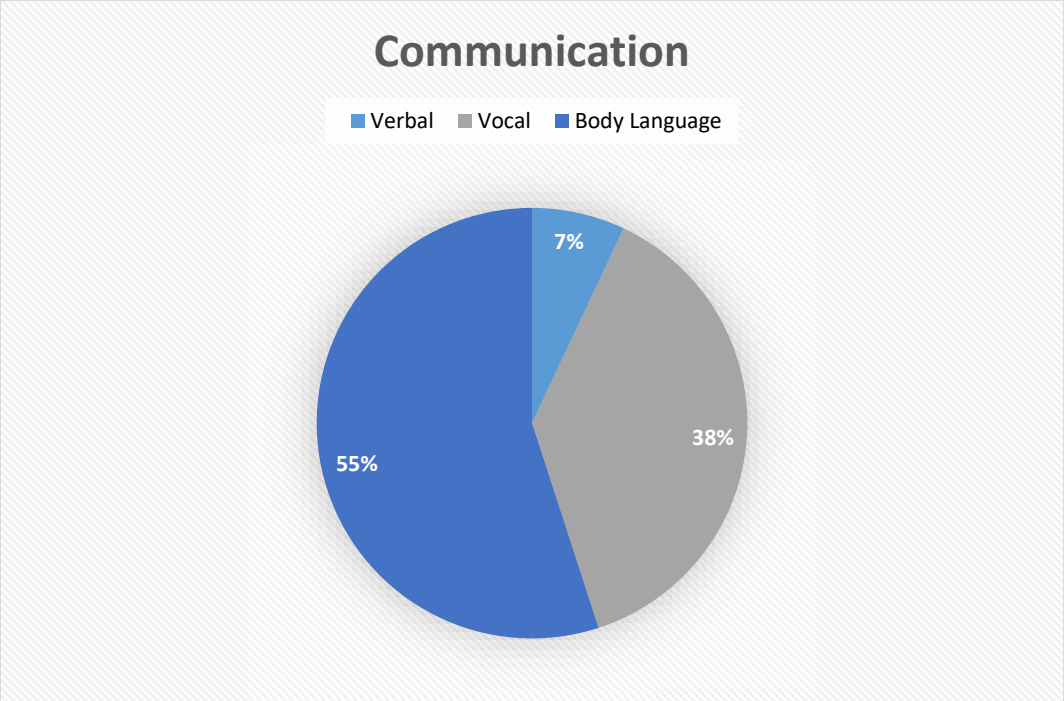

Figure 2.1. Percentages of communication

On the other fields, there have been extensive research in to detect emotions through voice tone, EEG signals, body gesture, speech, facial expression or, as the combination would work more efficiently, multimodal algorithms. Due to the extension of social media, people are posting various contents online. Hence, there are datasets based on social media apps [40] that are gathered through *YouTube* or models analyzing data on *Twitter* accounts.

Assuming a communication when both ends of the conversation are present, there are two major channels to convey an emotion: Visual and Aural channels. As Ekman et al. suggests that facial expressions display six basic emotions (Anger, Joy, Sadness, Disgust and Surprise) [41]. There have been various studies based on Visual datasets and facial signs that consider the Facial Action Coding System Ekman developed [49].

That is mainly the reason as to why Verbal and Vocal aspects of communication are neglected even though they play a considerable role in communication. Especially vocal which appears to be a vague approach towards conveying a meaning, is remarkably helpful as result of being global. Studies such as [42] argue how signal

processing behind voices could be useful, where subject is unable to speak properly (infant) or the body language and facial expressions are not necessarily meaningful or of any help.

2.4. Datasets

In order to facilitate emotion detection, simulate various algorithms, and achieve higher accuracies, constructing a reliable dataset is crucial. Datasets are distinguishable by their scope. The scope of each dataset contains several variables such as number, age, and/or gender of speakers, number of dialects (if any), languages, number and/or type of emotions, etc.

There are three types of (emotions in) datasets: Natural, Simulated, Elicited [21]. In that paper (as in many others), “Natural” is considered the type of dataset gathered during a continuous speech and developed on real data such as normal conversations (not necessarily in normal situations) between two or more people. “Simulated” type of dataset is defined as produced by professional actor or actress perform specific emotions. Elicited dataset is the type that is compiled on induced emotions in artificial emotional situation, e.g., when a host in a talk show causes the audiences fear or surprise.

Although elicited emotions are natural, there has been a concern on ethical grounds while the data would be more accurate if the emotions are not simulated but stimulated. Specifically causing fear in subjects is ethically questionable and illegal in many countries [43]. Moreover, there is no measurement on how sincere the elicited emotion is [44]. Therefore, this type of dataset is not wildly used as much as the other two. However, there are some situations where participants willingly succumbed themselves to a negative emotion (stress, fear, etc.), for instance in amusement parks, and they are gathered for the use of different projects.

Various datasets are generated for different types of emotion detection, e.g., Facial emotions (Visual), Vocal emotions (Audio), Sentiment analysis (Text). While facial emotions are considered universal; vocal emotions may differ based on culture, language, gender, or age. Hence, the scope of datasets for emotion recognition through voice needs to be designed carefully.

Table 2.3 presents some of the most well-known datasets that are available for public either free or through a subscription. The types are classified in two labels: Simulated and Natural; the latter includes Elicited dataset as well. As the availability of some datasets remains unknown, it is presumed that they might be academically (or commercially) available, if not public or free. The table is mainly focused on Audio datasets and some of the famous multimodal datasets are not listed. Among which there are HUMAINE [45], and eNTERFACE [46], both publicly available, includes Visual plus Audio data. Moreover, there is ICT-MMMO [47] and YouTube dataset [48], both available by request via an email to GiotaStratou, and includes Audio plus Text plus Visual data suitable for sentiment analysis.

Table 2.3. Datasets

Name	Type	Access	Language	Size	Emotions
Berlin emotional database (EMO-DB) [49]	Simulated	Public and Free	German	800 utterances 5 males 5 females	Anger, joy, sadness, fear, disgust, boredom, neutral
Danish emotional Database [50]	Simulated	Public with license fee	Danish	2 males 2 females	neutral, surprise, happiness, sadness, anger
Interface [51]	Simulated	Commercially Available	English, Slovenian, Spanish and French	English:8928 (two males, one female) Slovenian:6080 Spanish:5520 French:5600 sentences (One male, one female for each)	Anger, disgust, fear, joy, surprise, sadness, slow neutral, fast neutral
The AIBO database [52]	Natural	-	English and German	Children	anger, bored, emphatic, helpless, ironic, joyful, motherese, rest, reprimanding, surprise, touchy

FERMUS III framework [53]	Simulated	Public with license fee	German and English	2829 utterances males 21 1 female	Anger, disgust, joy, neutral, sadness, surprise
SUSAS [54]	Simulated – Natural	Public with fee	English	16,000 utterances 13 females 19 males	Stress, noise, fear, anxiety, angry, neutral, depression
LDC (Linguistic Data Consortium) Emotional Prosody Speech and Transcripts [55]	Simulated	Commercially available	English	actors 8 2433 utterances	Neutral, panic, anxiety, hot anger, cold anger, despair, sadness, elation, joy, interest, boredom, shame, pride, contempt
BabyEars [56]	Natural	Private	English	509 utterances 6 males 6 females	Approval, attention, prohibition
Swedish company Voice Provider [57]	Natural	Academically available	Swedish	61078 (Telephone Service Customers)	Neutral, emphatic, negative
Interactive Emotional Dyadic Motion Capture (Audio-Visual) [58]	Simulated	-	English	5 males 5 females	happiness, anger, sadness, frustration, neutral
Surrey Audio-Visual Expressed Emotion (SAVEE)	Simulated	Public and Free	English	4 actors 480 utterances	Anger, disgust, fear, happiness, surprise, sadness, neutral
Toronto emotional speech set (TSSE) [59]	Simulated	Public and Free	English	2800 utterances 2 females	anger, disgust, fear, happiness, pleasant surprise, sadness, neutral

Montero et al. [60]	Simulated	Available	Spanish	2000 phonemes per emotion – 3 passages, 15 sentences of neutral content	Happiness, sadness, cold anger, surprise
Amir et al. [61]	Natural	-	Hebrew	19 males 21 females	Anger, fear, joy, sadness, disgust
Marc Schroder [62]	Simulated		German	6 native speakers (3 males, 3 females)	Admiration, threat, disgust, elation, boredom, relief, startle, worry, contempt, anger
Makarova and Petrushin [63]	Simulated		Russian	12 males 49 females 10 sentences per emotion 3660 utterances	Anger, fear, surprise, happiness, sadness, neutral
DEMoS (Database of Elicited Mood in Speech) [64]	Elicited	Available free of charge	Italian	23 females 45 males 9,365 emotional and 332 neutral samples	anger, sadness, happiness, fear, surprise, disgust, neutral, guilt
Scherer et al. [65]	Natural	-	English, German and French	100 males (25 German, 16 English, and 59 French speakers) produced about 100 sentences of read speech and several passages of spontaneous speech	Stress and load level
Tato et al. [66]	Elicited	-	German	14 non-actor speakers (7 males and 7 females) 2800 utterances	Anger, boredom, happiness, sadness, neutral

Caldognetto et al. [67]	Simulated	-	Italian	One native speaker	Anger, disgust, fear, joy, sadness, surprise
Lee and Narayanan [68]	Natural	-	English	Customers and call attendants	Negative (anger, frustration, boredom), Positive (neutral, happiness, others)
The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) [69]	Simulated	Commer cially and academically available	English	12 females 12 males 7356 files	Speech: Calm, happy, sad, angry, fearful, surprise, and disgust expressions Song: Calm, happy, sad, angry, and fearful emotions
Wu et al. [70]	Simulated	-	Chinese	25 males 25 females 30 speakers: Evaluation Dataset – Short passages and 20 command phrases 20 speakers: each 20 utterances for each type of emotion	Anger, fear, happiness, sadness, neutral
EMA (Electromagneti c Articulography) Database [71]	Simulated		English	10 sentences for each emotion (one professional female and one non-professional) 14 sentences for each emotion (one non-professional male)	Dimensiona l: valence, activation, and dominance (Anger, happiness, sadness, neutral, other)

				680 sentences	
GEES [72]	Simulated	Open	Serbian	Three females Three males 2790 recordings and three hours duration of speech	Anger, happiness, sadness, fear, neutral
Fernandez and Picard [73]	Natural		English	Four drivers 598 utterances (154, 156, 137 and 151 for each subject)	Stress
Grimm et al. [74]	Natural	Publicly available	German	12 hours of Audio-Visual recording from TV talk show (Vera am Mittang) 44 males and 60 females (between 16 and 69 years old)	Dimensional: a. Activation (calm-excited) b. Valence (positive-negative) c. Dominance (weak-strong)
PDREC (Persian Drama Radio Emotional Corpus) [75]	Simulated	Available	Persian	15 females 18 males 748 utterances	Anger, boredom, disgust, fear, happiness (joy), sadness, surprise, neutral
ANAD (Arabic Natural Audio Dataset) [76]	Natural	Available and free of charge	Arabic	Eight videos of live calls between an anchor and a human outside the studio from online Arabic talk shows	Anger, happiness, surprised
EMOVO [77]	Simulated	Available for scientific and research purposes	Italian	Three males (age 30, 27 and 30) Three females (age 28, 23 and 25)	Anger, disgust, fear, joy, happiness, sadness, surprise, neutral

				588 recordings	
The LEGO emotion database [78]	Natural	Available	English	347 dialogues with 9,083 system-user exchanges	garbage (non-speech, critical noisy recordings or just silence), non-angry, slightly angry, very angry
UUDB (the Utsunomiya University Spoken Dialogue Database of paralinguistic information studies) [79]	Elicited	Public	Japanese	12 females 4840 utterances	Dimensional: pleasant–unpleasant, aroused–sleepy, dominant–submissive, credible–doubtful, interested–indifferent, positive–negative

In this section, a brief introduction to some of most popular datasets is presented. However, before that, it is worthwhile to point out the shortcoming that becomes obvious after going through Table 2.3 which is the absence of public and free datasets of other most spoken languages, such as Mandarin and/or Chinese. Moreover, if a researcher is willing to work based on theories believing in the fact that emotions are learned not innate; they would lack a proper database including various languages, for, in that case, emotions may appear differently by the change of languages.

2.4.1. The AIBO database [52]

AIBO is a cross-linguistic (German and English) natural database, generated through human (children)-robot interaction. German data is collected from 51 children (age: 10-13; 21 males and 30 females) instructing a Sony's AIBO dog using a map. English version of this data is generated through the same procedure by 30 children between the ages of four to fourteen.

In several experiments, children believed the robot was listening to them, while it in fact it was following a programmed scenario in order to provoke different reactions.

Several experienced labellers studied the pauses and intonations and assigned following labels to the collected data: “Neutral, joyful, surprised, emphatic, helpless, touchy (irritated), angry, motherese, bored, reprimanding, rest (non-neutral, but not belonging to the other categories)”.

Recordings of the German database took place in two classrooms in which four persons were present: a child, a supervisor, a wizard (pretending doing the recording), and a third assistant. The English dataset (including two groups of E1 and E2) were collected in a multi-media studio in CETADL (the Centre for Educational Technology and Distance Learning) in the Department of Electronic, Electrical and Computer Engineering (EECE). In both cases, there are also video recordings which are only for internal use due to privacy restriction.

The English version contains 5822 words from the vocabulary of 247 words with the average of 85.7 and 135.5 words per session for E1 and E2. The German data, on the other hand, included 51,393 words from a vocabulary of 1190 words (841 real words, 349 non-words).

The characteristics of recording equipment are as follows: two head-mounted wireless microphones: UT 14/20 TP SHURE UHF-series with microphone WH20TQG, a Senheiser ew100 range lapel microphone (SK100 transmitter, EK100 receiver) clipped to the SHURE head-mount. An existing wall-mounted microphone in CETADL is also utilized. For the purpose of analogue to digital conversion, the Edirol UA-5 external sound card along with USB interface is used. The sample rate is 44.1 kHz.

2.4.2. Toronto emotional speech set (TESS) [88]

TESS is widely known and utilized. It is a simulated dataset generated by two actresses (26 and 64 years old) who spoke English as their first language. The actresses are educated and have musical training. They portrayed seven emotions of anger, disgust, fear, happiness, pleasant surprise, sadness, and neutral by saying a set of 200 target words and generating a dataset of 2800 utterances. Audiometric testing indicated that both actresses have thresholds within the normal range. The authors of this dataset are Kate Dupuis and M. Kathleen Pichora-Fuller and it is published from University of Toronto, Psychology Department.

This dataset is available for public free of charge.

2.4.3. Berlin Emotional Database (Emo-DB) [49]

This simulated database is generated by 10 actors equally represented by both sexes. The participants were selected by open advertisement in newspapers, as the researchers doubting the naturalness of emotions performed by real actors. However, after selecting ten performers among 40 volunteers, based on the naturalness and clarity of their utterances, it turned out that all but one of them passed acting schools. This explains the reason behind the popularity of using professional actors/actresses for generating simulated datasets.

Between discrete and dimensional type of models of emotions, researchers selected discrete type due to its clarity by both performer and listener. The model contains these emotions: Anger, joy, sadness, fear, disgust, boredom, and neutral.

For the text material, researchers consider two requirements: The sentences must be interpretable in the emotions under study, nonetheless, not emotionally biased. They indicated that two types of sentences would fulfill both requirements, either nonsense words or sentences, or normal sentences which could be used in everyday life. As the first one would be difficult for performers and lead to stereotyped overacting, the latter is selected. They relied on “the performers” ability of self-induction by remembering a situation when the desired emotion had been felt strongly, which is known as the Stanislavski method”.

As a trial to avoid some other issues, later, 20 other subjects took part in the study, who was evaluating the samples. The utterances were randomly presented to them, and they had to decide the emotion and level of convincingness for each. Afterwards, utterances with recognition rate more that 80% and naturalness more than 60% were chosen for furthers analysis. It resulted to the loss of 500 hundred out of the whole 800 hundred utterances.

In the end, two label files in ASCII format were created: the first one is “a narrow phonetic transcription which is based on the additive judgment supported by visual analysis of oscillogram and spectrogram”. The second one contains “segmentation into syllables and markings of four different levels of stress (sentence stress, primary,

secondary stress, unstressed)". Eight trained phoneticians gathered the second label in order to make the data more reliable.

This database is free and publicly available.

2.4.4. SUSAS [54]

The name of SUSAS (Speech Under Simulated and Actual Stress) provides the information that it is partially natural and partially simulated. The dataset is divided into five domains: talking styles (slow, fast, soft, loud, angry, clear, question), single tracking task or speech produced in noise (Lombard effect), dual tracking computer response task, actual subject motion-fear tasks (G-force, Lombard effect, noise, fear), psychiatric analysis data (speech under depression, fear, anxiety; Not Released though).

Speaking styles contain various speaking formats mentioned above. Single tracking task is when the speech data produced by performing a single response. The dual tracking task is recorded as one performs compensatory and acquisition tracking task, actual subject motion-fear tasks domain attempts to simulate the sudden change in altitude/direction experienced in an aircraft cockpit, and the last domain includes speech of patients under "psychiatric analysis in a natural doctor-patient environment".

The set of approximately 16000 utterances is generated by 19 males and 13 females with the age range of 22-76.

The simulated part of this dataset includes ten stressed styles (talking styles, single tracking task and Lombard effect domains) and the natural section contains speech under either one of these conditions: "dual-tracking workload computer tasks, or subject motion-fear tasks (subjects in roller-coaster rides)".

This dataset is publicly available but not free of charge.

2.5. Feature Extraction and Classification

Although auditory recognition of speech is based on three specific characteristics of one's voice (Pitch, Loudness, Timbre), emotion recognition through voice would be possible by extracting various features and combining them, i.e., there are two phases

regarding this procedure: first, feature extraction and second, feature classification. There are specific requirements such as signal processing, de-noising, and segmentation to deliver meaningful units of voice for classification.

In the general area of speech processing (speech recognition, speaker recognition, language recognition, and emotion detection by voice), the popular algorithms include Pitch detection, Energy detection, Formants, Mel Frequency Cepstral Coefficients (MFCC), Teager energy operated (TEO)-based features, Log Frequency Power Coefficients (LFPC) and Linear Prediction Cepstral Coefficients (LPCC).

These features may vary regarding the application, as in some models, speech rate or the duration of silence is considered. However, they do not necessarily depend on the speaker or lexical information.

Nevertheless, there is no agreement on the best feature for emotion recognition. These features are categorized in various ways. Some may categorize them by the High- and Low-level features, or local and global features which are the specific ones. Prosodic features extracted from each segment (frame) are local features, and global features are statistics extracted from the whole utterance. While global features excel at accuracy and speed as their number is less than local features, they could only distinguish between high- and low-arousal emotions. In other words, the overall form of high-arousal signals (Joy or Anger) is alike. Also, using global features, temporal information would be lost which makes the global features unsuitable for complex classifiers such as Hidden Markov Model (HMM) or Support Vector Machine (SVM). Global features lack the sufficient number of training vectors for these classifiers.

Having mentioned the prior labels, according to [80], there are other specific categories, one of which is the following four: continuous features, qualitative features, spectral features, and TEO-based features. The best result, supposedly, is achieved through a combination of features from all four categories.

2.5.1. Continuous Features

Continuous features, such as pitch or energy related features, are believed to acquire most of the emotional content from speech. Williams et al. expresses that the arousal (high or low activation) directly effects the energy, its distribution across the

frequency spectrum, frequency itself and duration of pauses across the whole signal [81]. The fact is the reason continuous features are widely used in the emotion recognition field. However, the advantages and disadvantages of this group of characteristics are similar to global features. This is not surprising, considering some of the most commonly used global features categorized among continuous features. For example, as there are similarities between fear and joy in fundamental frequency (F0), they are often mistaken for each other using F0 for classification.

In overall, continuous features are grouped into four other categories themselves: Pitch-related, formants, energy-related, timing and articulation features.

2.5.2. Voice Quality Features

Voice quality features, like voice level or temporal structure, come alongside the full-blown emotions which are strong, in opposition to underlying emotions that would not cause an action, but affects it in a negative or positive way. Phonetic variables are the main contributors to the impression of voice quality. Cowie et al. suggested the acoustic correlators in voice quality area fall into four groups: Voice level, voice pitch, temporal structures and phrase, words, and feature boundaries [82].

However, not many researchers favor measuring the “quality” of voice as there is no agreement upon their actual meaning. Talking about quality, one may use labels such as harsh or breathy, which might be interpreted differently from person to person. In order to determine whether an utterance is tense, researcher must be creative as there is no specific ruler. Hence, there would be more disagreement and less accuracy.

In Table 2.4, a summary of variation in voice different features is presented. There are only a subset of emotions and features, though. Moreover, the observation may vary according to one’s personality, culture, gender, or many other factors.

Table 2.4. Features’ variation based on emotions [21]

Emotions	Pitch	Intensity	Speaking Rate	Voice Quality
Anger	Abrupt on Stress	Much Higher	Marginally Faster	Breathy, Chest
Disgust	Wide, Downward inflections	Lower	Very much faster	Grumble, Chest tone

Fear	Wide, Normal	Lower	Much faster	Irregular voicing
Happiness	Much wider, Upward Inflections	Higher	Faster/Slower	Breathy, Blaring tone
Joy	High mean, wide range	Higher	Faster	Breathy, Blaring timbre
Sadness	Slightly Narrower	Downward Inflections	Slower	Resonant

Also, another comparison based on [83] is presented in Table 2.5 which although is quite similar, as the main purpose of that article was studying on human emotions, the table appeared to be more complete.

Table 2.5. Feature's variation based on emotions (2) [83]

Emotion / Feature	Anger	Happiness	Sadness	Fear	Disgust
Rate	Slightly faster	Faster or slower	Slightly slower	Much faster	Very much faster
Pitch average	Very much higher	Much higher	Slightly lower	Very much higher	Very much lower
Pitch range	Much wider	Much wider	Slightly narrower	Much wider	Slightly wider
Intensity	Higher	Higher	Lower	Normal	Lower
Voice Quality	Breathy, chest	Breathy, blaring tone	Resonant	Irregular voicing	Grumble chest tone
Pitch changes	Abrupt on stress	Smooth, upward inflections	Downward inflections	Normal	Wide, downward terminal inflections
Articulation	Tense	Normal	Slurring	Precise	Normal

2.5.3. Spectral Features

Spectral features are based on frequency domain. They are a short-time representative for speech signal and extracted through the transformation of a time-based signal into the frequency domain using Fourier Transform, as the emotional load

of an utterance has an impact on the spectral energy distributed across a range of frequencies [84].

Spectral features are acquired by various approaches [80]: Linear Predictor Coefficient (LPC), One-sided Autocorrelation Linear Predictor Coefficient (OSALPC), Short-time Modified Coherence Method (SMC) [85] and Least Square Modified Yule-Walker Equations (LSMYWE) [86]. However, to fully extract the distribution over the audible range of frequency, the spectrum is filtered by a bank of band-pass filters, which their bandwidths' usually follow a non-linear scale such as Bark scale or the famous Mel-frequency scale.

There has been a constant debate on whether the Cepstral-based features derived from linear features while using a linear predictor (LPCC, or OSALPCC) is better than linear based features when it comes to emotion recognition or not. MFCC outperformed in specifically stress detection [86], up until [84] introduced the new algorithm using Hidden Markov Models (HMM), comparing two types of features (linear- and cepstral-based): LFPC, the linear-based feature and LPCC and MFCC, the cepstral-based features. LFPC, by the average accuracy of 77.1%, stands on top while MFCC and LPCC with 59.0% and 56.1% accuracy respectively clearly fall behind.

2.5.4. Non-linear Teager Energy Operator (TEO)-based Features

Non-linear TEO-based features are specifically powerful means to detect stress, even though they totally fail in speech recognition and stress classification area. Teager suggested that as speech produced by a non-linear airflow through vocal system, and muscles tend to tense under stress which influences the vocal system, a non-linear feature is required for stress detection [87]. Teager [88] and Kaiser [89] introduced the Teager-Energy-Operator, based on the assumption that "hearing is the process of detecting energy". In TEO and for a discrete time signal, x[n] is defined as:

$$\Psi\{x[n]\} = x^2[n] - x[n-1]x[n+1] \qquad (2.1)$$

Where $\Psi\{\}$ is the energy operator in discrete time and x[n] is the nth signal component.

They noticed that under stressful conditions, not only the fundamental frequencies but also the harmonics distribution over the critical bands changes [87]. Since the TEO of multi-frequency signals would reflect the interaction between each frequency components, TEO-based features could be used for stress detection task.

However, while TEO-based features outperform other features in stress detection (in [90] with two assumptions that: 1) The speech is not free-style, and 2) The text is available.), they give poor accuracies for speech recognition and classification (Loud, Angry).

Table 2.6. Some Features and their classifiers

Authors	Features	Classifier	Database	Accuracy
Nicholson et al. [91]	Prosodic features	One-class-in-one Neural Networks	Large database of a phoneme balanced words	Around 50%
Ramamohan and Dandapat [92]	10 frequencies (ascending order sorted) and 10 phase angels corresponding to ten most significant peaks of amplitude from positive to negative	VQ-HMM		80% (Telugu) 70% (English
Koolagudi et al. [93]	Statistical features (mean duration of each utterance, mean and standard deviation of pitch values, mean energy across each utterance	Euclidean distance classifier	IITKGP-SESC	75% (male) 69% (female)
Petrushin [94]	Pitch, the first and second formants, energy, speaking rate	Real-time emotion recognition using NN	Call center application	77%
Iliou and Anagnostopoulos [95]	Prosodic features (pitch, energy, duration, etc.)	NN	Berlin emotional speech corpus	51%

Nwe et al. [84]	Log-frequency power coefficient (LFPC)	Discrete Markov Model	10 sentences: Anger, fear, joy, sadness, disgust and surprise in Mandarin and Burnese	76.4% (Mandarin) 79.9% (Burnese)
Kandali et al. [96]	MFCC, tfMFCC (TEO-in-transform-domain MFCC), LFPC, tfLFPC, WPCC (Wavelet Packet Cepstral Coefficient), tfWPCC, WPCC2 (composed by method 2), tfWPCC2	GMM	MESDNEI (5 languages)	73.1% 75.1% 60.9% 61.1% 95.1% 93.8% 93.7% 94.29%
Firoz Shah et al. [97]	Discrete Wavelet Transforms (DWTs) MFCCs	ANN	Elicited database containing 700 utterances in Malayalam	68.5% 55%
Ververidis et al. [98]	87 calculated features over 500 utterances using The Sequential Forward Selection Method (SFS) resulting 5-10 best possible features	Bayes classifier	Danish Emotional Speech database	61.1% (male) 57.1 (female)
Vogt and André [99]	1280 features consisting of pitch, energy, MFCCs, pauses, durations, speaker rates Reduced to 90-160 by correlation-based feature selection method	Naïve Bayes Classifier	Berlin emotional speech corpus	69.1% (full features) 77.4% (selected features)
Kwon et al. [100]	Log energy, formants, F0 information, mel-based energy, MFCCs (with velocities), acceleration coefficient	QDA SVM HMM LDA	SUSAS database AIBO database	96.3% (Stressed/neutral style) 70.1% (four-class speaking style with Gaussian SVM) 42.3% (AIBO, 5 class emotions, speaker-independent)

Harimi and Emayilian [101]	Prosodic and Spectral features	LDA	PDREC (Persian Drama Radio Emotional Corpus) Berlin emotional speech corpus	55.74% (female) and 47.28% (male) 78.64% (female) and 73.40% (male)
Alonso et al. [102]	Two prosodic features and four paralinguistic features (related to pitch and spectral energy balance)	SVM	EMO-DB LDC Polish database	94.9% 88.32% 90%
Lin and Wei [103]	39 features extracted and compared to MFCCs performance by SFS, best subset selected	HMM SVM	Danish Emotional Speech Database	98.9% (female), 100% (male) and 99.5% (gender independent) 89.4% (female), 93.6% (male) and 88.9 (gender independent)

After the introduction of different types of features, Table 2.6 provides some of the works on feature extraction and classification with the accuracies they achieve. Additionally, a brief introduction to some of the most popular features in voice and speech area is offered in following. Not just any feature suits the purpose of emotion recognition through voice signal or speech recognition (e.g., MFCCs are popular in the field of Automatic Speech Recognition), nevertheless, they are all studied.

- **LPCCs (& LPCs)**

Among the several types of features generated from autoregressive (AR) models, there is linear predictive coding coefficients (LPCCs). LPCCs are used to capture emotion-specific information manifested through vocal tract features, as they provide accurate estimates of spectral characteristics of a speech signal [104].

Linear predictive methods are mostly used for speech and speaker recognition, or speech synthesis, and quite close to the FFT. The envelope is calculated from several formants or poles specified by the user. They remove redundancy of signal and try to predict the next point as a linear combination of previous values and is appropriate for modeling vowels which are periodic, except nasalized vowels.

The process of computing and extracting LPCCs is presented in Figure 2.2. As it is shown, the stages start with the block framing of speech signal, windowing after that and then autocorrelation analysis fallowed by LPC analysis which results LPC. Converting the LPC parameter leads to extracting LPCC.

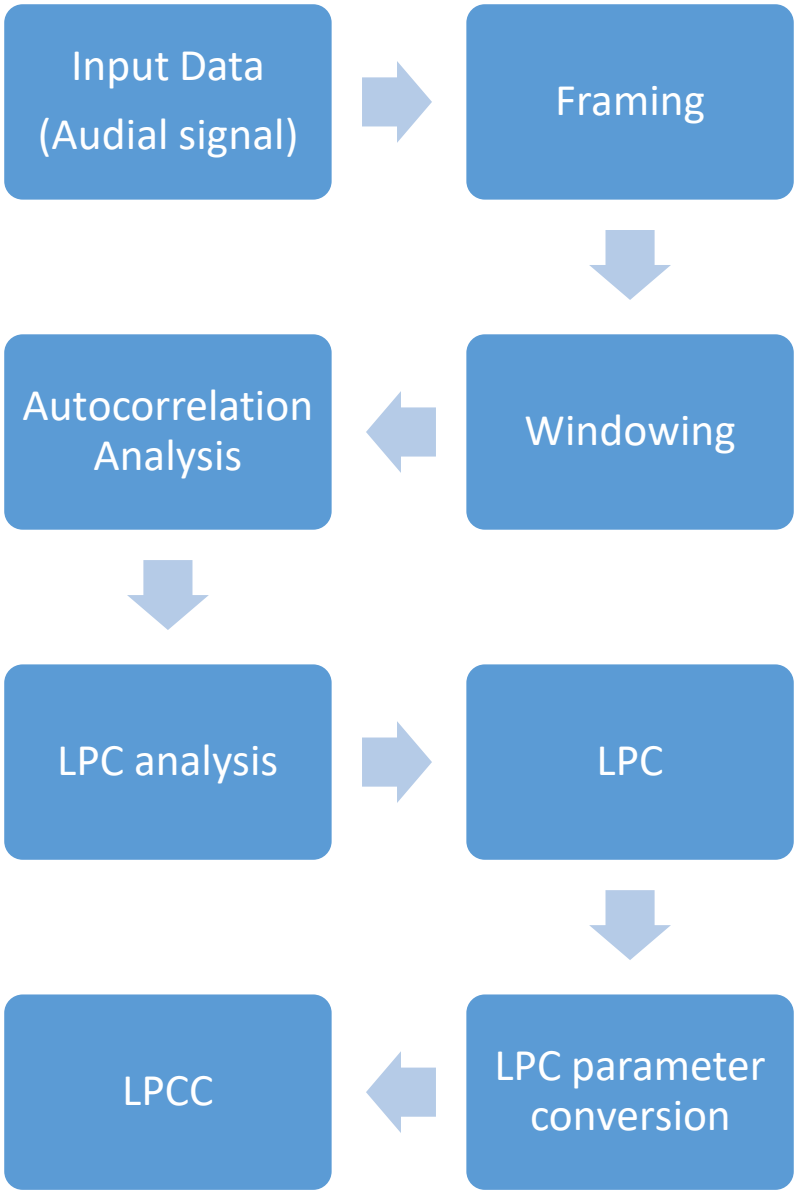

Figure 2.2. Flowchart of LPCC calculation

LPC is based on the source-filter model of speech signal and the order of an LPC model is the number of poles, or formants in the filter. Two poles for each formant are the usual. Added two to four poles are to represent the characteristics of source. The

order is related to the sample rate of the audio signal: 10000 Hz - LPC order = 12-14 (males) and 8-10 (females); 22050 Hz - LPC order = 24-26 (males) and 22-24 (females).

The number of poles can be calculated by the equation below.

N Poles = SampleRate/(F0max*0.25) (2.2)

Although algorithms developed based on this feature would achieve a very high level of accuracy, it is only if the speech is recorded in a clean environment.

Algorithm: *LPCC bsed ELM Classifictation*

Input: A training dataset $\mathbf{A} = \{\mathbf{x_i}, \boldsymbol{y_i}\}_{\boldsymbol{i=1}}^{\boldsymbol{N}}$;
Activation function $\boldsymbol{g}(\ )$
$\boldsymbol{L}$ *as the hidden number*;
Query sound signal frame
Procedure Training( )
begin

$generate\ \{w_j, b_j\}, j = 1,2, \dots L\ randomly$;
calculate H as the hidden layer output matrix as in [105];
obtain the output weight $m = H^t T$;
end
Procedure Testing( )
begine
using the linear compination of past instant samples(1)*and LPCC features order*(2)*to calculate LPCC vector* x_q

[105];

Calculate the hidden node vector of x_q *as in* (3);
find the label l as in formula(4);
end

(1) $s(n) = q1s(n-1) + q2s(n-2) + \cdots + qps(n-p).$

(2) $$\begin{cases} c_1 = q_1 & \\ c_n = q_n + \sum_{k=1}^{n-1} \frac{k}{n} c_k q_{n-k} & if(1 < n \le p) \\ c_n = q_n + \sum_{k=1}^{n-1} \frac{k}{n} c_k q_n - k & if(1 < n \le p) \end{cases}$$

(3) $h(x_q) = \left[g(w_1.x_q + b_1), \dots g(w_L.x_q + b_L)\right]$

(4) $o_q = h(x_q)\beta$

- **MFCCs**

MFCCs first were introduced in 1980's by Davis and Mermelstein for the purpose of identifying monosyllabic words [106]. It is based on Mel (after the word "Melody" [107])

scale, which is a unit of pitch equals to one thousandth of the pitch of a simple tone with the frequency of 1000 Hz and amplitude of 40 dB above the auditory threshold. Since MFCC carries a remarkable amount of information about the physical aspects of the speech signal and its structure, it is popular among speaker and speech recognition researchers [108]. MFCCs would be calculated in seven steps as below [109]:

1) Input signal is pre-emphasized,
2) Short-time Fourier analysis is performed,
3) Magnitude spectrum is calculated,
4) Wrap it into mel-spectrum using 26 triangular overlapping windows (Center frequencies are equally distributed on the mel scale),
5) Take the log operation on the power spectrum,
6) DCT application on the result so the cepstral features are derived,
7) Perform cepstralliftering.

These steps are summarized and presented below, by Figure 3.12.

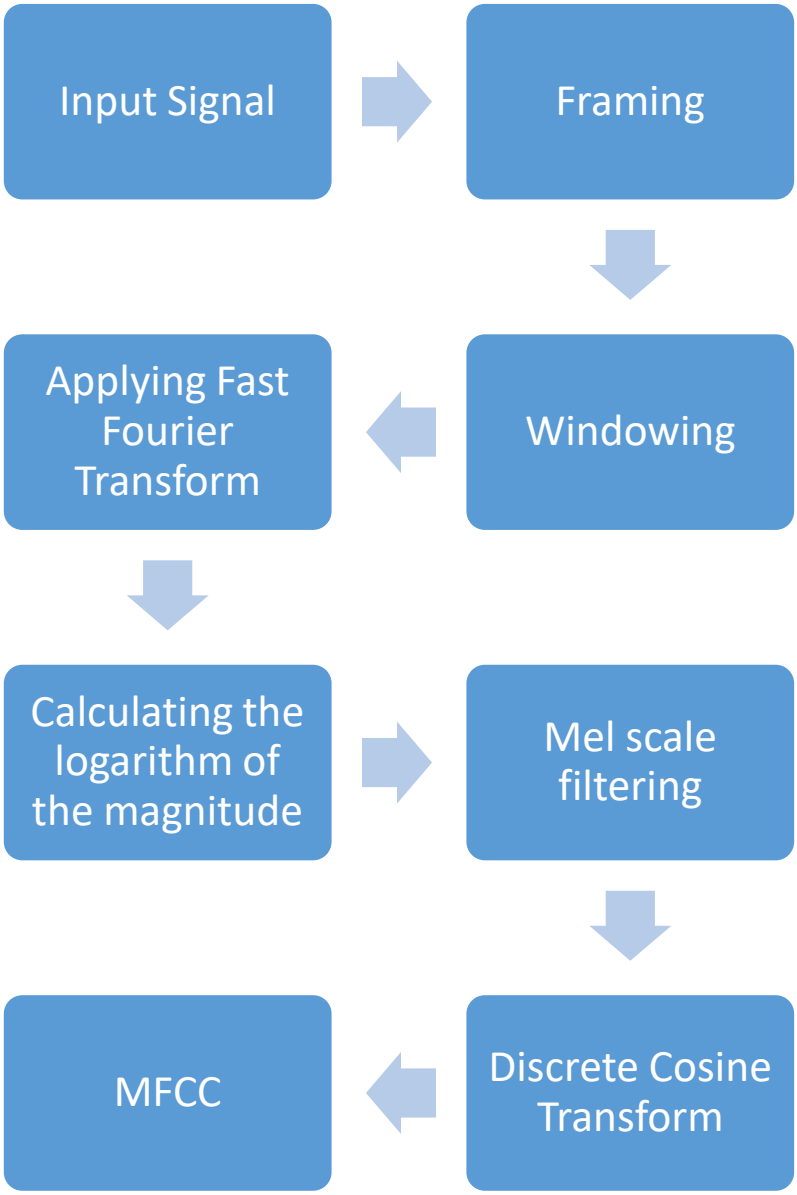

Figure 2.3. A block diagram of a MFCC extraction process

Shortly after the initial introduction, various improvements have been applied which are mainly lied within the change of number and shape of filters. According to [110], the most popular MFCCs are:

MFCC FB-20 – introduced by Davis and Mermelstein, (1980), HTK MFCC FB-24 – from the Cambridge HMM Toolkit (HTK) described by Young et al. (1995), MFCC FB-40 – from the MATLAB Auditory Toolbox developed by Slaney (1998), HFCC-E FB-29 (Human Factor Cepstral Coefficients) – proposed by Skowronski and Harris (2004).

A simple pseudo code for MFCC extraction would be like this:

Algorithm: *MFCC calculation*

Input: *audioIn "/∗ A simple tonw with the frequency of* $100Hz$ *∗/*
Procedure*MFCC*(*parameters*)
begin

initialize parameters;
split *phonocardiogram signals* ***into frames***;
apply ***Hamming windowing*** *to the frames*;
computing ***short − time*** *fourier transform for all frames*;
determine martix for a mel − spaced filterbank;
transform the spectrum into a mel spectrum;
compute ***MFCC vector*** *for each frame by applying* ***DCT***;

end
output: *Mel Frequency Cepstral Coefficient*

- **GFCCs**

The reason of MFCCs poor performances on noisy data could be the use of triangular filters to model the critical bands. To overcome this issue, instead of the triangular filters, GammaTone filters are proposed, and the extracted features are called gammatone frequency cepstral coefficients (GFCCs) [111] which are based on the GammaTone Filter Bank (GTFB). Their vectors are calculated from the spectra of a series of windowed speech frames. GFCC could be extracted in four steps as below [109]:

1) The input signal goes through 64-channel GTFB
2) At each channel, absolute value is taken and decimated to 100 Hz as a way of time windowing. This creates a time-frequency (T-F) representation that is a variant of cochleagram [112].
3) The cubic root of the T-F representation is taken.
4) Discrete Cosine Transform (DCT) is applied to derive cepstral features.

This process and steps are summarized in Figure 2.4.

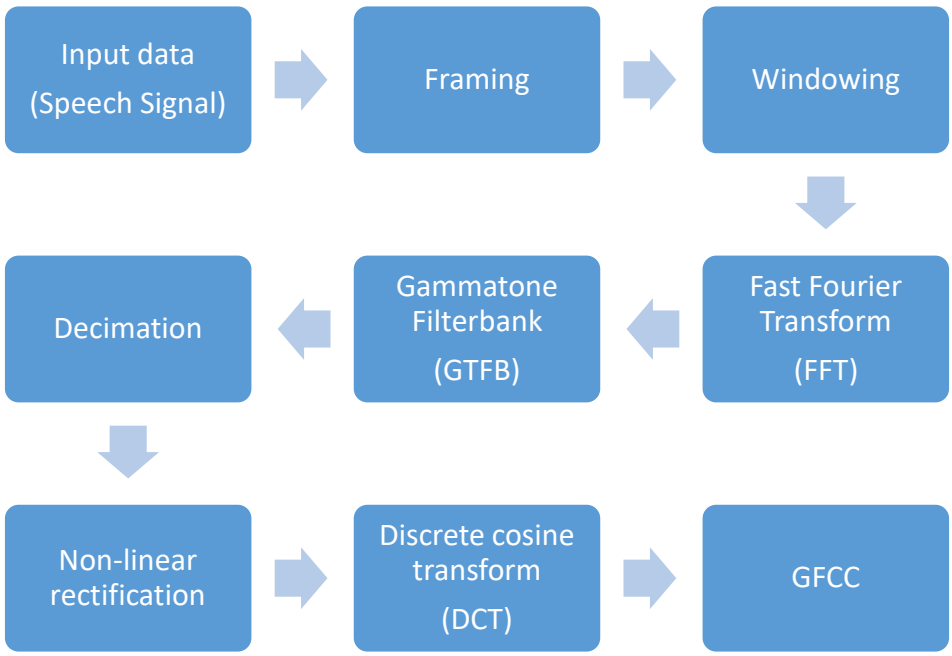

Figure 2.4. GFCC extraction process

GFCC has finer resolution at low frequencies than MFCC. The pseudo code for GFCC extraction would be as below:

Algorithm: *GFCC*

---

input: *input signal audioIn*
output: *GFCC*
Procedure GammatonCoef()
begin
do
GammaFilterBank(audioIn)
foreach channel
do
*decimated the absolute value of the channel to 100Hz /*This creates a time-frequency(T-F) representation that is a variant of cochleagram*/*
end
take the cubic root of the values
calculate the Discrete cosine transform to the cepstral features
end

As it is clear from observations above, there is no perfect feature even for a specific type of classifiers. It is recommended that after considering the type of task and classification, suitable features are selected accordingly.

2.6. Deep Learning

In the last few years, the new challenges and applications pointed to deficiencies of traditional machine learning algorithms since the tasks expected rapidly became more complex. Machine Learning techniques are capable spotting the minute differences while categorizing data; however, that appeared to be the problem. In cases such as image processing, the computers need to disregard some differences and regard some others. The algorithms need to define a set of essential features (e.g., pitch, speed, pause, etc. in voice processing) and be insensitive to irrelevant ones (e.g., accent, gender, etc.). As a result, a subset of ML came into existence by the name of Deep Learning (aka. Deep Neural Network). The major difference between Deep Learning and the classical ML methods is that Deep Learning is an artificial intelligence capable of learning and improving by unstructured unlabeled data. This is the reason Deep Learning methods are considered: the ability to learn automatically and find common features within a category. It is the closest AI to an actual human brain wiring which works the same to some extent, however, not as efficient, and powerful.

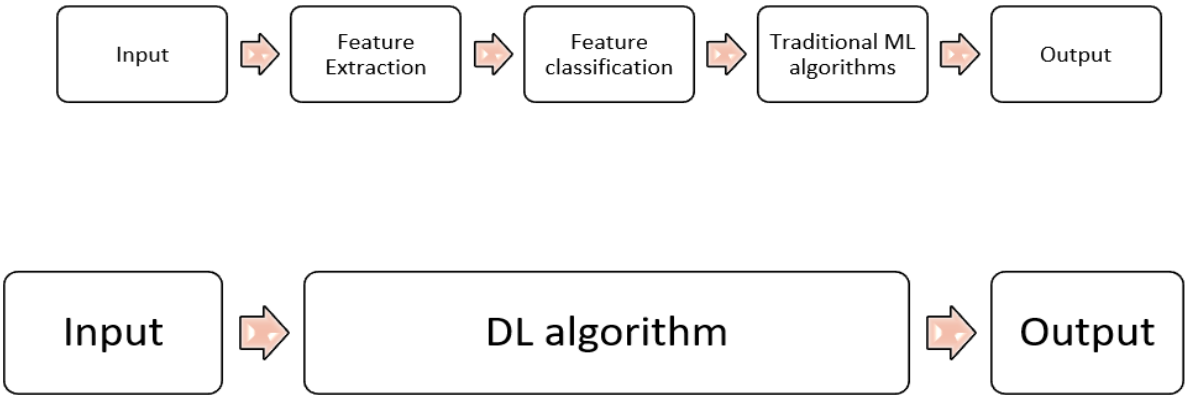

Figure 2.5. Comparison between DL networks and traditional ML networks

As it is shown in Figure 2.5, the difference between DL algorithms and traditional machine learning algorithms is that deep learning models perform feature extraction and classification through their learning process, hence, many problems due to poor feature selection method in the algorithm would perish. This is if not the main, one of the most significant advantages of deep learning methods.

It is not only in theory, but also empirically, deep learning methods are shown to result better accuracies in fields such as speech emotion recognition, or image processing. In Table 2.7 as presented, there are obvious results proving the supremacy of a deep neural network in terms of accuracy when it comes to emotion recognition through voice.

Table 2.7. Comparison between various Classifiers in Speech Emotion Recognition [113]

Algorithm	Anger	Happy	Sad
k-nearest neighbor	93%	55%	77%
Linear Discriminant Analysis	63%	49%	72%
Support Vector Machine (SVM)	74%	70%	93%
Regularized Discriminant Analysis	83%	73%	97%
Deep Convolutional Neural Network (DCNN)	99%	99%	96%

Other than outperforming traditional methods in speech emotion recognition field, deep learning models delivers promising results in medical fields such as reconstructing brain networks [114] or studying quantitative structure-activity relationship in possible drug molecules [115]. As LeCun, Bengio and Hinton [116] predicted, it is safe to assume deep learning field has a promising future ahead due to its obvious advantages over traditional methods.

By 1980s, researchers started to work on one specific task for each program, divided into various categories: Speech Recognition, Chess Playing, Image Processing, and so on. Moreover, launching world-wide-web into the public domain in 1991 was an unnoticed step up for Deep Learning, since, nowadays, internet provides models and algorithms with considerable amount of data, making them able to even learn on their own.

Having that all said about the importance of data in training Deep Learning algorithm, once Deep Mind, later acquired by Google, introduced AlphaGo Zero in 2017, there was a new approach coming to existence and new window opening. AlphaGo Zero, specialized in playing the ancient Chinese game is trained solely by playing itself. The Zero in its name stands for “Zero Data”, as it receives no data other than the basic rules of the game. It defeated AlphaGo, the champion of the time, became the world’s best AI player. It concludes clearly: the future lies within self-training models.

However, in recent years, considering disadvantages of supervised learning and complication of self-training, Neural Networks with unsupervised learning methods such as Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) are most considerable, the former is usually used for image processing purposes and the latter, specifically Long Short-Term Memory (LTSM) networks, is suitable for speech recognition or music composition.

There are also Feedforward Neural Networks such as Autoencoders, Multilayer Perceptron, Radial Basis Function Networks, but other than various NNs, there is Generative Deep Learning Algorithms one of which is Deep Belief Networks (DBN), the most famous one. DBN is also wildly used for image and video recognition [112].

In terms of learning methods, there are three types of learning algorithms for machines, depending on the task they are expected to perform. The first type of learning, Supervised Learning, is used when the desired output is labelled and known to the user. In this type of learning, user defines the correct answer for machine and algorithm calculates its errors and tries to train itself through labelled data. Examples of algorithms that work as supervised learning algorithms are Decision Tree, k-nearest neighbor algorithm (KNN), Random Forest, etc. Although supervised learning algorithms are easy to generate and implement, they are time consuming and demand heavy (though simple) engineering of data.

Then, there is a second type of learning, unsupervised learning. This type does not require desired output to be labelled, as there is no predicted output. The types of algorithms work through unsupervised learning are suitable for clustering population of data. It is mostly suitable for dividing customers into different groups for advertising and recommendation purposes. The algorithm trains itself by the feedback it receives from

the user, deciding if its clusters are set correctly or not. K-means and Apriori algorithm are the example of such algorithms. The algorithms using unsupervised learning do not require labeled data or the type of heavy engineering that first type demands. However, they require huge amount of data which is why they were not common until very recently, the era of online public data.

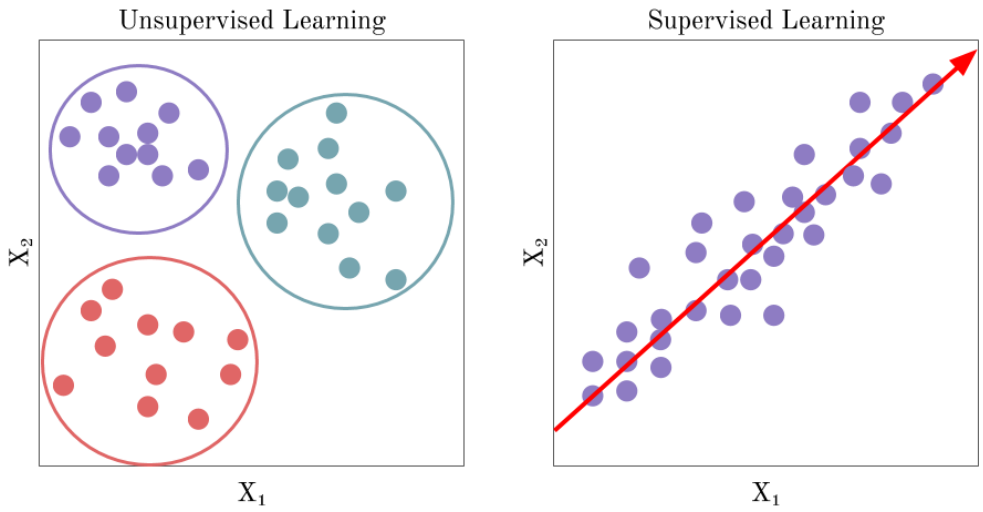

Figure 2.6. Unsupervised Learning Vs. Supervised Learning [117]

The last type of learning which recently has come into existence is Reinforcement Learning [114]. In this approach towards learning, machine is exposed to environment and train itself by trial and error. An example of this type of learning is mentioned in Deep Learning section, Alpha Zero, in which Zero stands for “Zero Data”. In other words, this type of learning does not require any data other than the policy of that environment or, if playing a game, rules of the game. The task is to reach a certain point and a specific goal, while updating after sending an action to environment and receiving an observation alongside with a reward. The new generation of GPU and powerful processors make it possible for this type of algorithms to perform their tasks. Reinforcement Learning can be formulated as Markov Decision Process [132].

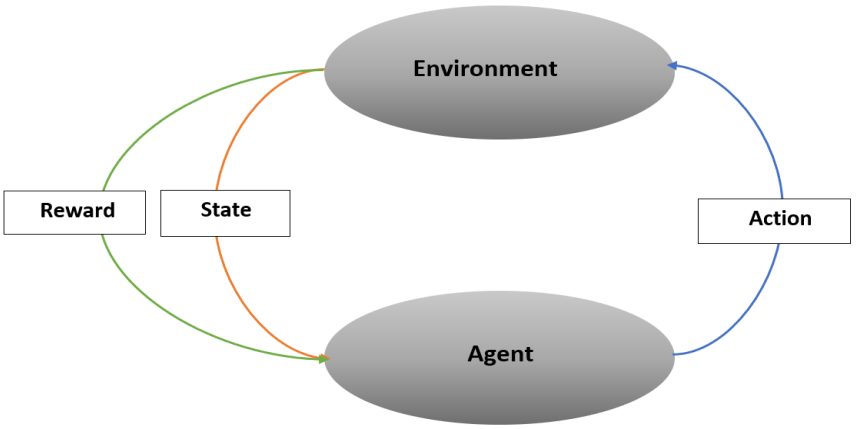

Figure 2.7. Reinforcement Learning

The algorithms below each could do supervised, unsupervised or reinforcement learning. These types only define the way the algorithms learn and train, not their structure. In other words, one can implement any kind of algorithm to be trained in any way necessary, either the output is known or not.

A neural network algorithm, as previously mentioned, is based on human-brain neural network. It is emerged from a concept Frank Rosenblatt developed during 1950s and 1960s, inspired by Warren McCulloch and Walter Pitts. The concept is known as Perceptron; hence, deep feedforward neural networks sometimes are referred to as Multilayer Perceptron as well.

Neural network originally (at the most basic level) possesses an input layer, a hidden layer, and an output layer. To calculate the value for output layer, each input for a neuron is weighted and summed, before adding a bias to the summed value. Afterwards, the signal is processed by an Activation Function and generate the output.

A Neural Network and its different types do not necessarily belong to a Deep Learning algorithm. There are two conditions need to be satisfied for a Neural Network to be Deep Neural Network. The first one is that the network needs to have more than three layers, assuming input and output layers are considered as one layer. When there is more than one hidden layer, the Artificial Neural Networks would change to Deep Neural Networks.

The second condition is yet relevant to the first one: A nonlinear activation function is required. In order to understand the reason, consider Figure 2.8.

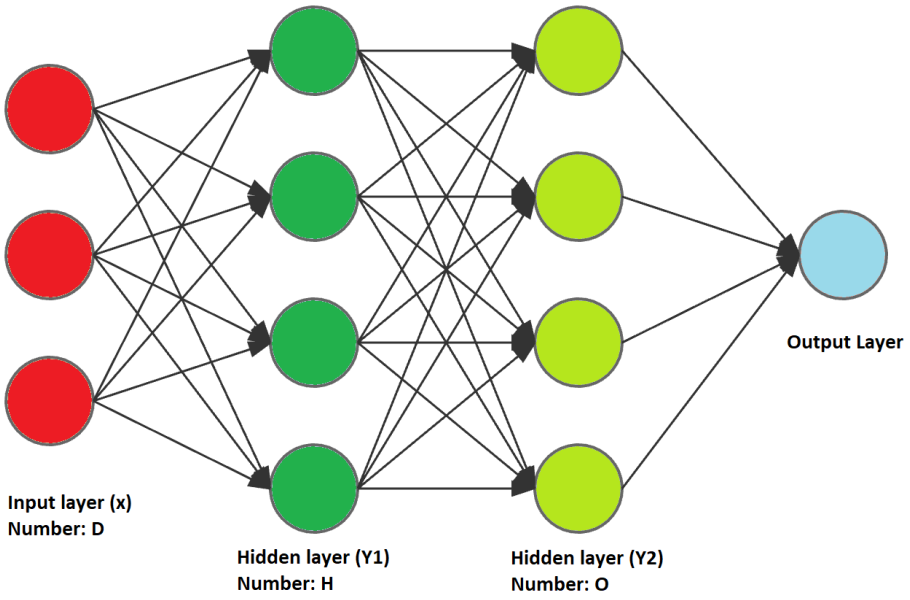

Figure 2.8. A Neural Network with a linear activation function

Given the activation function for the first hidden layer is φ and for the second hidden layer is γ, the equations would be as follows.

$$Y_k^2 = \gamma\ (\sum_{j=1}^{H} w_{kj}^2 Y_j^1) \quad (2.3)$$

$$Y_j^1 = \varphi\ (\sum_{i=1}^{D} w_{ji}^1 x_i) \quad (2.4)$$

If the activation function φ is linear, it means:

$$\varphi(z) = z \quad (2.5)$$

Hence:

$$Y_k^2 = \gamma\ (\sum_{j=1}^{H} w_{kj}^2 (\sum_{i=1}^{D} w_{ji}^1 x_i)) \quad (2.6)$$

In other words:

$$Y_k^2 = \gamma\ (\sum_{i=1}^{D}\ (\sum_{j=1}^{H} w_{kj}^2 w_{ji}^1) x_i) \quad (2.7)$$

Assuming:

$$\sum_{j=1}^{H} w_{kj}^2 w_{ji}^1 = V_{ki} \quad (2.8)$$

The output would be:

$$Y_k^2 = \gamma\ (\sum_{i=1}^{D} V_{ki}\, x_i) \qquad (2.9)$$

V is only a new weight, a function of *k* and *i*, which means the number of layers does not affect the outcome as it would be only the result of inputs. Thus, the network is not deep. Therefore, the activation function needs to be non-linear.

At the present, there are various non-linear activation functions among which there are tanh(z) or 1/(1+exp(-z)), the smooth activation functions. However, Glorot et al. proved that Rectified Linear Unit (ReLU) activation function which is a half-wave rectifier f(z) = max(z,0) usually learns faster when it comes to networks with more than three layers [118]. This activation function makes unsupervised learning methods more feasible, without the need of a pre-training.

2.6.1. Feedforward Neural Networks

Feedforward Neural Networks are named such since the information only flows forward. The difference between these types of networks (e.g., Convolutional Neural Network, Auto-encoders, Radial Basis Function Neural Network, etc.) and Recurrent Neural Networks is that they do not have cycles or send feedbacks backward. The term Feedforward is in fact referring to the method of the network for learning.

Also, while it is safe to say all Multilayer Perceptron algorithms are feedforward, the opposite is not necessarily true since some feedforward algorithms do not possess multilayers, hence, are not deep.

In this type of Neural Network, Feedforward networks, a method named Back Propagation (BP) is utilized in order to train the network. The brief explanation is that BP algorithm computes the gradient of the loss function in network considering its weights, then, updates the weights. Even though it is calculating the gradient backward, does not affect the Feedforward nature of the network since the data still moves forward only, and the calculation is used for later updates of the weight.

As a group of researchers gathered under the name of Canadian Institute for Advanced Research (CIFAR) around 2006 and proposed unsupervised learning methods which learns features without requiring extensive effort to label the data [119].

These feature detectors training methods, also known as *Pre-training*, reduces the weight of the neural networks to a sensible value and later, using a standard back propagation, the whole deep algorithm would be fine-tuned. According to LeCun et al., the primary use of this approach was in the field of speech recognition, and it became possible as there have been improvements regarding computing power and processors [119]. The Graphics Processing Units (GPUs) which have been replacing CPUs in recent years, are fast to process and easy to program, which is the reason it is possible to train more heavy networks nowadays.

In 2017, Fayek et al. combined a feedforward network with Recurrent Neural Network to recognize four emotions alongside silence from speech [120]. The technique achieved 64.78% accuracy applied on IEMOCAP database, showing there is a possibility of future work in this field with the attempt of recognizing more than Anger, Happy, Sad, and Neural emotions.

Auto-encoders (AEs)

Auto-encoder is an unsupervised feedforward learning technique which reconstruct the unlabeled input data, based on probabilistic graphical models. The performance of this algorithm improves through minimizing the *Reconstruction Error*, resulted by the difference between the original data and the generated replica. The nature of this type of learning method requires a balance to be maintained: although the model is supposed to be sensitive enough to reach an acceptable accuracy reconstructing the data, it needs to be insensitive so that it would not simply memorize the input.

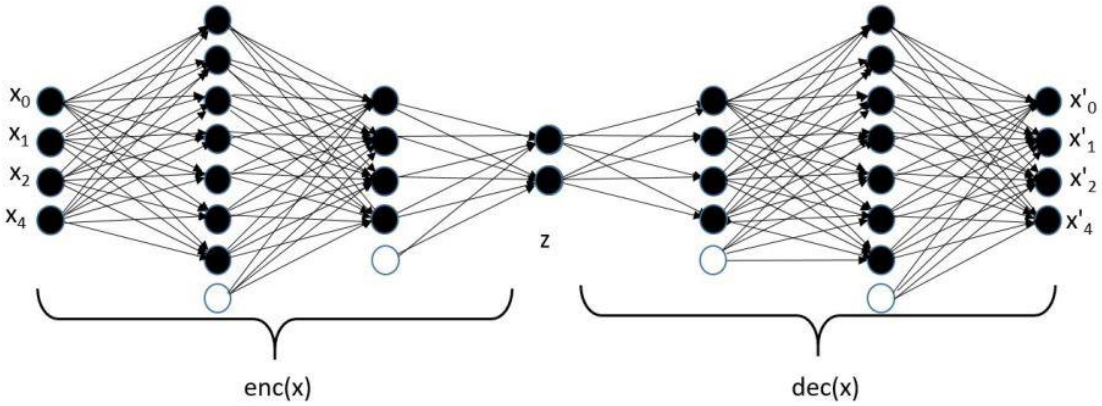

Figure 2.9. Autoencoders Neural Network [121]

The AEs, as shown in Figure 2.9, consist of two sections: Encoder and Decoder, enc(x) and dec(x). Given the weight *w* and the biased term *b*, the model works as indicated below:

$$X \rightarrow \text{enc}(X): z = \sigma(wX + b) \qquad (2.10)$$

$$\text{dec}(X) \rightarrow X': X' = \sigma'(w'z + b') \qquad (2.11)$$

And the Reconstruction Error would be calculated as:

$$L(X, X') = ||X - X'||^2 = ||X - \sigma'(w'(\sigma(wX + b)) + b')||^2 \qquad (2.12)$$

The nature of AEs makes them suitable to be applied in single layers of other neural networks. For example, one might use an AE in hidden layers of their RNN or CNN so that improves the accuracy and eliminate the drawbacks common among those networks. Hence, there are variations of AEs to be utilized accordingly. Some of the most famous types are Sparse Auto Encoder (SAE which encourage sparsity and responds to the unique input data); Variational Auto-Encoders (VAE which are generative models similar to Generative Adversarial Networks and make assumption regarding the data distribution by the use of log-likelihood); Denoising Auto-Encoder (DAE which their main task is to recover a corrupted data fed to the network), etc.

In 2018, Sahu et al. applied an Adversarial Auto-encoder technique on IEMOCAP database in order to experiment its ability to extract 1582 features for speech-based emotion recognitions using OpenSMILE toolkit [136]. Using Unweighted Average Recall (UAR) as the evaluation metric resulted 57.88% accuracy, showing promises for further works on this technique.

Also, Eskimez et al. combined a Convolutional Neural Network with four kinds of unsupervised learning methods: DAE, VAE, AAE and Adversarial Variational Bayes (AVB) on USC-IEMOCAP audio-visual dataset [123]. They proved all performances improved with UAR comparing to hand-crafted features and achieved 47% of accuracy on F-1 score. The recognized emotions are Anger, Frustration, Neutral and Sadness. There might be significant improvements with the use of Recurrent Neural Networks in future works, considering the fact that VAE, AAE and AVB outperformed DAE.

Convolutional Neural Networks (CNNs)

Another feedforward neural network is Convolutional Neural Networks (CNNs) which is inspired by primate visual cortex, hence, suitable for image and video processing or other pattern recognition tasks. Even though it originally meant for processing visual data (rather than the data in the form of audio), in [124] a CNN is used which possess three fully connected layers and extracts emotional features of a speech signal and by turning them into spectrogram images, classifies them.

According to [116], there are four features which distinguish CNNs from other forms of neural networks: “local connections, shared weights, pooling and the use of many layers”.

The general architecture of CNNs is a simple combination of several (repetitive) stages: convolutional layers followed by pooling layers. The formers extract features from initial data by filtering them through small windows, computing dot product between the weight and the original input; pooling layers on the other hand, merge the extracted information which are similar to each other into one. There are various types of pooling layers each suitable for different tasks (e.g., max pooling, average pooling, etc.). Before pooling and after a convolutional layer has extracted the features, a non-linear activation function, usually ReLU, is applied.

Once there is more than one sequence of convolutional layers and pooling layers, the CNN is a deep network, DCNN. As the information flows from bottom to top, the low-level features are primary ones such as edge detectors.

On the top of the network, a classifier is applied so that the features are classified by it, might be SVM or Softmax which calculates the probabilistic regarding different classes and scores each.

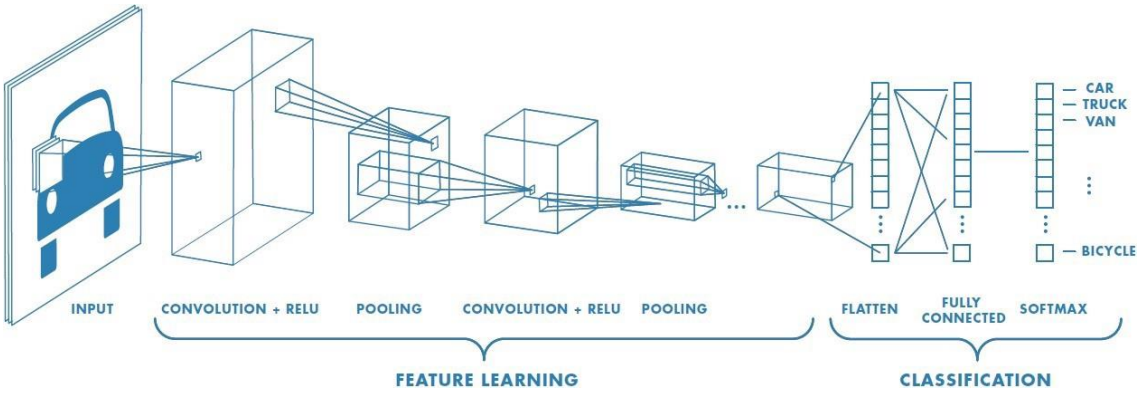

Figure 2.10. Convolutional Neural Network [125]

CNNs basically are designed for image and video recognition. However, it is possible to convert an audio signal to an image or through other approaches use a CNN for related tasks such as Natural Language Processing (NLP) or speech processing. For example, Zhao et al. used a deep CNN in order to learn features for emotion recognition task for seven different emotions consisting of Anger, Disgust, Fear, Happiness, Joy, Neutral and Sadness [126]. The merged DCNN applied on Berlin Emotional Database and IEMOCAP resulted a 92.71% accuracy learning from audio and log-mel spectrograms.

Also, a LTSM based CNN was applied on CMU-MOSEI database in [127] for multimodal emotion recognition through both speech and text to recognize Anger, Disgust, Fear, Happiness, Sadness and Surprise. Achieving 83.11% accuracy for speech emotion recognition (SER), the model seems promising for further work.

2.6.2. Recurrent Neural Networks (RNNs)

Recurrent Neural Networks are the branch of Neural Networks in which information flows both forward to the next layer, and backward to provide feedback for the previous layer as a method of learning. They are most suitable for the tasks requiring sequential input such as language or speech as they are capable of predicting the future state of the input due to their interdependency. Prihodko et al. believes that the reason that RNNs considered suited for emotion recognition through speech signals is their short time framing for voice features [128].

As an example, Mirsamadi et al. proposed a Deep Recurrent Neural Network to recognise Anger, Disgust, Fear, Happiness, Neutral, Sadness and Surprise on

IEMOCAP database and compared the result to traditional SVM algorithms [129]. The deep RNN outperformed the traditional method with +5.7% absolute improvement in weighted accuracy and +3.1% of improvement in unweighted accuracy.

Moreover, [130] tried a combination of RNN and CNN on IEMOCAP database for iCLub robot in order to detect Anger, Happiness, Neutral and Sadness. On in-domain data, the model achieves 83.2% accuracy.

However, even though RNNs proved to be powerful networks, there is one significant drawback regarding their performance which is the tendency in their backpropagated gradients to either vanish or explode. Sutton [131] indicates that even though it is the primary goal of an RNN, both theory and experience prove the difficulty of storing data for long.

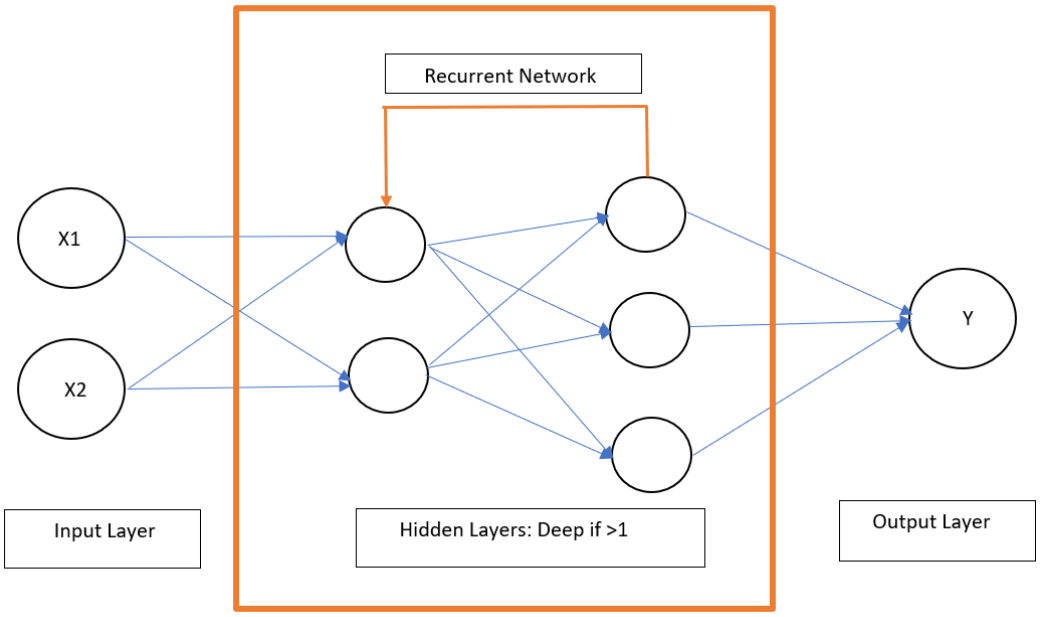

Figure 2.11. Recurrent Neural Network

In order to overcome the problem, a subset of RNNs is introduced by [132] named Long Short-Term Memory (LTSM) to improve the network. The nature of LTSMs makes it possible for them to remember the data for a longer period. There are hidden units implemented within the network with which accompanied weights are one while they are connected to themselves. Hence, the units tend to copy the original input in each step, leads to remembering them for long.

Tripathi et al. proposed a LTSM based RNN with three layers tested on IEMOCAP database to recognise Anger, Happiness, Neutral and Sadness [133]. It achieved 71.04% accuracy, providing a path to future works in which implementing extra layers may lead to higher accuracies.

Also, the deep 1D and 2D CNN LTSM network in [134] was applied on Berlin Emotional Database and IEMOCAP to detect Anger, Disgust, Fear, Happiness, Neutral, Sadness and surprise. The models achieved 91.6% and 92.9% accuracy respectively.

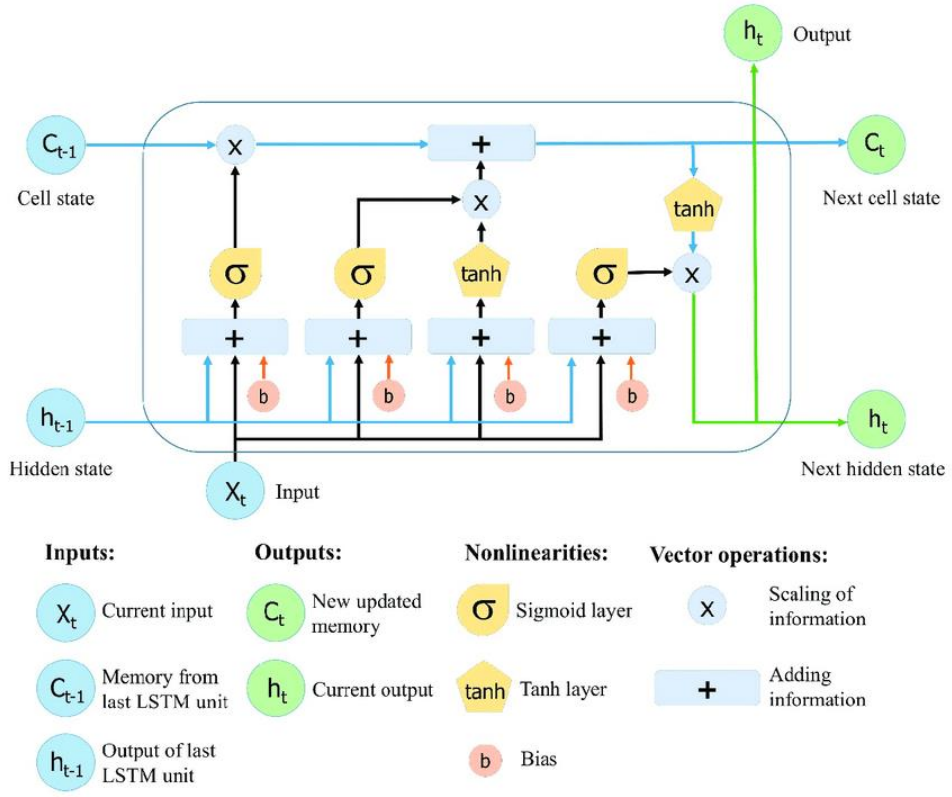

Figure 2.12. The structure of the Long Short-Term Memory (LSTM) network [135]

Other than LTSMs, advancement in architectures and the progress in ways to train RNNs, hybrid deep learning algorithms are paths towards more powerful networks. As Fayek et al. proposed a hybrid model. [136], a real-time speech emotion recognition algorithm, in which the convolutional layers are augmented with RNNs, the same course of action which is taken by [130] to improve human-robot interaction through emotion recognition. According to Khalil et al. this type of hybrid networks is able to acquire both temporal and frequency characteristics of a speech signal [21]. There are other types of

hybrid models where an AE is implemented within an RNN for the purpose of reconstructing voice features and later, the RNN predict the emotion.

2.6.3. Generative Adversarial Networks (GANs)

Generative Adversarial Network is a generative model most suited for applications using unsupervised learning, or reinforcement learning as in AlphaGo Zero.

Once training data is provided, GAN generates fake similar data so that convinces the discriminator model into classifying the fake ones as real, hence, increases the error rate of the classifier. The discriminator might be any simple normal classification model. This is how the model is trained in unsupervised mode or learning through reinforcement by playing itself. The point of generator model is that it extracts the features from real input so that it could properly produce fake data which is indistinguishable from the real one.

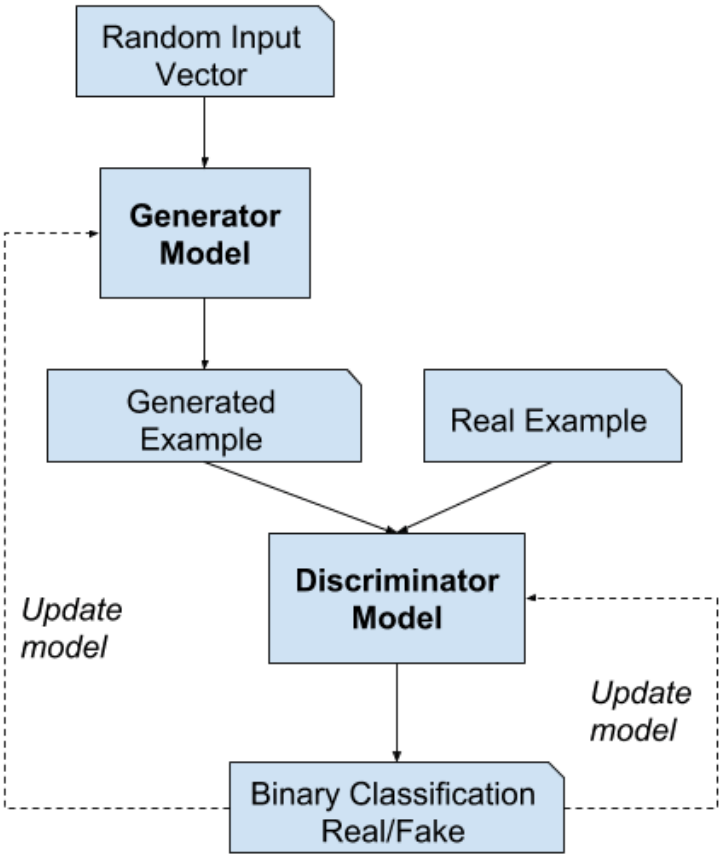

Figure 2.13. Generative Adversarial Network structure

The nature of this type of network (and its usual applications) suggests that it is most suited for image processing like Convolutional Neural Networks. Hence, they work well with CNN models as discriminators or generators. They could provide data and train CNNs for vision recognition and image processing tasks. They also have applications in video games (as they could generate fake people or scenes which seems real), science (as in [137] simulate the distribution of Dark Matter in a specific direction), etc.

2.6.4. Deep Belief Networks (DBNs)

Before understanding the concept of Deep Belief Network which is a generative model and a combination of several Restricted Boltzman Machines (RBM), it is necessary to unfold the structure of an RBM.

Restricted Boltzman Machine is a type of Boltzman Machine (which itself is a generative deep learning algorithm), and as it takes a probabilistic approach, it is also known as Stochastic Neural Network. In this category of network, two types of units are implemented: Visible units, and Hidden units. The restriction in connection between visible and hidden units makes it easy to implement them comparing to the original Boltzman Machine. While in Boltzman Machine all hidden and visible nods are connected to each other, in RBMs no two nods in the same layer are connected, hence, easier implementation. Once several RMBs are fine-tuned through back propagation and gradient decent, it would generate a deep belief network.

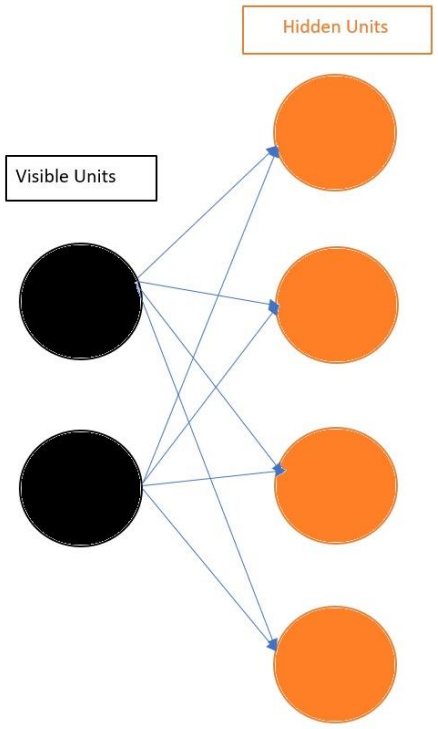

Figure 2.14. Restricted Boltzman Machine (RBM)

The multiple layers of RBMs in Deep Belief Network is pre-train phase of the network. Afterwards, the network is fine-tuned through normal feedforward learning process and back propagation. The simple structure of the network as Hinton first proposed in 2006 is presented in Figure 2.15. Since DBNs are less prone to vanishing gradient (comparing to RNNs), less complex in terms of computational aspects (hence, less expensive), and posses the ability to recognize patterns as in signals, they are suitable for speech emotion recognition tasks.

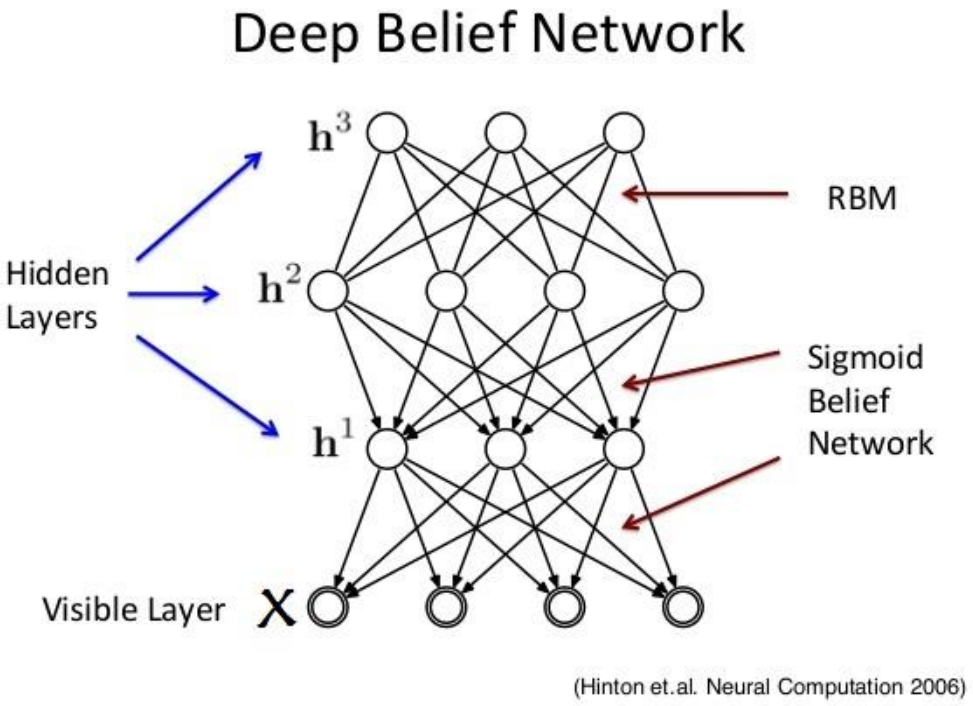

Figure 2.15. Deep Belief Network

The other advantage of DBNs is the unsupervised pre-training phase due to the nature of RBM layers implemented within the network. As a result, it is possible to train a DBN network with large amount of data even though not labeled.

In a comparison between a RBM based DBN and a SAE combined with SVM both applied on a variety of databases (FAU-AIBO, IEMOCAP, EMoDB, SAVEE and EMOVO) by [138], the DBN network outperformed the latter on all databases and languages, even slightly.

Before that, in 2017, there was another comparison carried out between DBN and SVM for Speech Emotion Recognition task on CAS emotional database by [139] to classify six emotions: Anger, Fear, Joy, Neutral, Sadness and Surprise. In this case, too, Deep Belief Network achieved a higher accuracy (94.6%) comparing to SVM model with the accuracy of 84.54%.

Moreover, there is a hybrid model, combination of SVM and DBN presented by [140] applied on a Chinese Academy of Sciences emotional speech database to detect seven emotions in Chinese language. The model delivered 94.6% accuracy and a future project could be applying it on databases of other languages.

Having all the advantages mentioned, there are also a visible drawback within the structure of this type of network. Comparing to RNN networks which re-adjust their weights since the information flows both ways, the greedy layer of DBN learns features based on the previous layer as the algorithm goes bottom-up. This disadvantage might lead to another network outperform DBNs where there are sufficient labelled data available.

2.7. Future Directions

The state of art in Speech Emotion Recognition is elaborated and studied in previous sections, one can see the blind spots in order to navigate the future path.

Starting from Emotion Models, the studies shows that even though emotions are divided into various categories and ranges, so far emotion recognition at its very best is limited to seven primary emotions: Anger, Disgust, Fear, Happiness, Neutral, Sadness and Surprise (or Joy in some cases). A future path could be not only progress in recognition of primary emotions, but also considering more inclusive emotion models such as [141], [142] or [143]. As comparisons in Feature section indicate, in order to recognize more emotions (and do so more accurately), it might result remarkable improvement that instead of determining proper features for all emotions, researchers search for the suitable features to detect each emotion separately.

Also, for advancement towards more complex emotions, the initial requirement is to obtain diverse datasets. As mentioned in Dataset section, most datasets are in English and highly limited to maximum seven emotions. This shortcoming makes it a challenging task to improve algorithms in a manner in which the algorithm recognizes various emotions regardless of user's language. Moreover, the vast majority of databases are simulated. There is an obvious need for Natural databases for testing models, hence one could observe how the model performs in real world scenarios.

Additionally, the potential future directions for different algorithms, specifically DNN algorithms, are mentioned in previous section. In this section, the suggestion and elaboration of the current state of art includes both traditional and more up-to-date methods for Speech Emotion Recognition (SER). It is not out of place to mention in recent years, SER witnessed a shift from supervised algorithms to semi-supervised learning as it is a more efficient approach towards Machine Learning and excludes the need of heavy labelling by humans. It is possible that as LeCun et al. predicted in 2015 [116], even though that Unsupervised Learning Methods has been outperformed by Supervised ones, in future they find their way back to the spotlight.

This prediction is supported by the reason traditional algorithms such as GMMs and HMMs are mostly abandoned ever since Deep Neural Networks came into existence. GMMs and HMMs both are powerful algorithms, first suitable to utilize acoustic features and the second more appropriate for “temporal variation in speech signals” [21]. However, they both needs manual labelling and a vast variety of data in order to achieve higher accuracies. That is why most of them require considerable amount of effort and time to appear successful. Moreover, in cases like [144] that have performed an experiment on DNN and GMM for the task of Emotion Recognition through voice, the results showed that implementing layers would improve the performance. As various features (MFCCs, PLPs, and FBANKs) are combined, the accuracy improved from 92.1% (single feature) to 92.3%. The study indicates that when a model is using a Deep Learning algorithm achieve higher accuracies in comparison to traditional classifiers like GMM.

On the other hand, although Deep Neural Networks overcome the issue of manually labelling data and being time-consuming, as mentioned previously, they demand implementation of multiple layers.

In order to solve both problems and achieve higher accuracies, Hybrid Models are being considered. For example, see [140] that is already discussed in previous section, is a combination of SVM and DBN and achieves 94.6% of accuracy in comparison to SVM alone, 84.54%, the same as DBM alone. As the model is implemented on CAS emotional speech dataset, there is a possibility to apply it on other languages and extend its application.

The urge to combine different algorithms to cover the other's flaws is not limited to combination of traditional methods and neutral networks. For example, since CNNs are originally for vision related tasks and RNNs have sensitivity regarding to vanishing (or exploding) gradients, they are one the most commons to be combined. As discussed in previous sections, some RNN models are augmented with layers of CNN (e.g., [145]) in order to overcome the drawbacks of both. Additionally, Sahu et al. used an enhanced memory reconstruction in a Recurrent Neural Network with Encoder and Decoder as in AAE to extract features and detect emotions respectively [146]. Similar models could be considered in future to improve the performance of RNNs.

2.8. Conclusion

The purpose of this chapter is to present different emotion models, features, datasets, techniques, and in overall, any knowledge necessary for Emotion Recognition, concentrated on voice modality. The section includes, and not limited to, the emotion models to which researchers are referring and tries to elaborate the psychological aspects of the work as well as its technical programming aspect.

Since Deep Neural Networks has been the center of attention in recent years for such tasks, my focus has been introducing various Deep Neural Network algorithms alongside citing the papers and models for future direction. Throughout the chapter, the drawbacks and shortcomings of each and any technique or dataset is pointed as a suggestion for potential future improvements. This chapter also provides an evaluation on the performances of various algorithms in the form of their accuracy percentage. The comparison between algorithms or the performance of a model on various dataset and/or through various feature extraction methods, showing the methods, datasets, and features suitable for each other, helps distinguishing the promising paths for improvements in Emotion Recognition by voice.

In the next chapters, one of the aforementioned datasets is selected and some of feature extraction methods are utilized in order to detect seven primary emotions alongside Neutral state from voice by both a MLP classifier with single hidden layer and a deep neural network. Each method is presented in a separate chapter and would be followed by a comparison and conclusion in the end.

Chapter 3.

Emotion Recognition through Voice using MLP Classifier

3.1. Introduction

In general, traditional machine learning classifiers fall into either of these two categories: Linear, and non-linear classifiers. There are many algorithms generated and developed so far such as GMM, HMM, SVM, Bayes Classifiers, KNN, etc. A few of the most popular methods are presented in Table 3.1 alongside the respective references and categories, i.e., if they are linear or non-linear. In some cases, like SVMs or ELMs, they could be linear **or** non-linear.

Table 3.1. some SER Classifiers with the references and linearity

Classifiers	**Linearity**
Bayes [1]	Linear
K-Nearest Neighbourhood (KNN) [1]	Linear
Gaussian Mixture Model (GMM) [2]	Non-Linear
Hidden Markov Model (HMM) [3]	Non-Linear
Support Vector Machine (SVM) [4]	Non/Linear
Extreme Learning Machine (ELM) [1]	Non/Linear

The type of exploration in the field of emotion detection varies from one study to another. Some studies such as those reported in [5] and [6] focus on delivering specific results with specific classifiers. Neiberget al. provides 85% of accuracy with GGMs applied on two sets of data and three categories of emotions [5], whereas Saketh et al. delivers 81% of accuracy through SVM with seven different emotions [6]. On the other hand, there are studies such as [7] that even though have implemented one classifier

(HMM), they provide accuracies for visual data (70%), audial data (30%) and bimodal data (72%) for classifying six primary emotions, anger, dislike, fear, happiness, sadness and surprise.

Another type of studies rises from comparison and combination of models. Petrushin conducted a study in order to compare a KNN classifier and a neural network in terms of accuracy while categorizing 5 emotions [8]. The KNN achieves 55%, ANN delivers 65% of recognition accuracy which improves by ensemble networks to 70% accuracy. There are further steps that studies like [9] would take in order to improve the performance of the model. In this study, for example, an ELM algorithm is augmented by the use of a Deep Neural Network and its accuracy increased from 45% for an HMM based model to 54.3% when the proposed model is applied to categorize 5 classes of emotions.

A trend starts to show up through conducted studies and approaches commonly taken. In order to reach higher accuracies, the complexity of models increase, thus, computations are more time-consuming. In other words, different algorithms work better with different features, hence, hybrid systems come into equation to improve the overall performance. For example, as GMM algorithms mostly demonstrate acoustic features and HMMs focus on temporal features in the voice signal [10]. They could be fused through a hybrid system such as Multilayer Perceptron (MLP) classifier for better results. Advancements in hardware area support more computationally complicated methods, however, there is a possibility to obtain acceptable results through more simple approaches.

In other words, the question is how far it is possible to simplify an Emotion Recognition algorithm without sacrificing efficiency. The presented study is not only focused on simplifying a method for Speech Emotion Recognition and showing moderate performance, but also is committed to demonstrate the limitation of a Multilayer Perceptron Classifier without the help of other classifiers as its layers. In Figure 3.1 the schematics of a MLP classifier is presented. Since the classifier is capable of possessing more than one hidden layer, it is possible to add various algorithms in its layers through which the model processes the data more efficiently. In this study, the model possesses only one hidden layer and total of three layers: Input,

hidden, and output layer. It receives three features and the labels for classes as input and delivers seven categories of emotions.

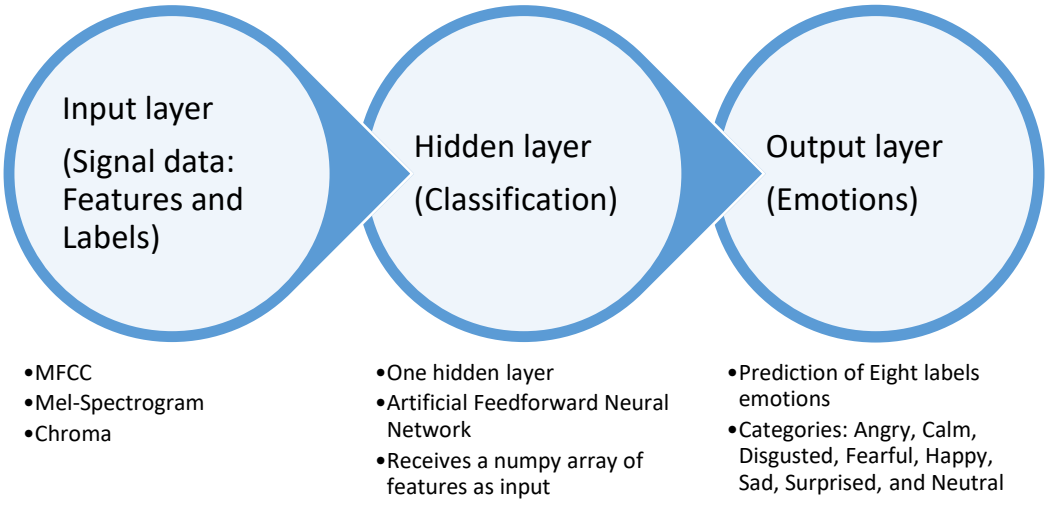

Figure 3.1. an illustration of a Multilayer Perceptron classifier

In this chapter, we study the above Artificial Neural Network, the dataset on which the algorithm is applied, and the features considered for the emotion recognition task. The initial aim of this thesis is to examine two methods against each other. I picked the Multilayer Perceptron Classifier as a traditional machine learning method as one of the algorithms. The objective of this chapter is to observe the performance of the first algorithm, a MLP classifier with a single hidden layer and compare its accuracy when changes are applied. Moreover, at the end of this chapter the limitations of this method will be discussed.

I also use the same dataset and features for both in order to have a more reliable comparison solely between the performances of the models. The coding is presented line by line through figures and a schematic of dataset, features and how they work or are processed is also explained throughout the chapter.

Towards the end of this chapter, confusion matrices and accuracies are provided to compare the performance of MLP classifier based on different feature extraction methods. The comparison of results leads to an analysis of the model's performance and if the features sit well with the MLP classifier. The chapter is concluded by remarks about the simulation studies.

3.2. Dataset

The dataset for training the algorithms is the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) available for public free of charge in [11]. Only the audio data is used as the study is focused on speech emotion recognition.

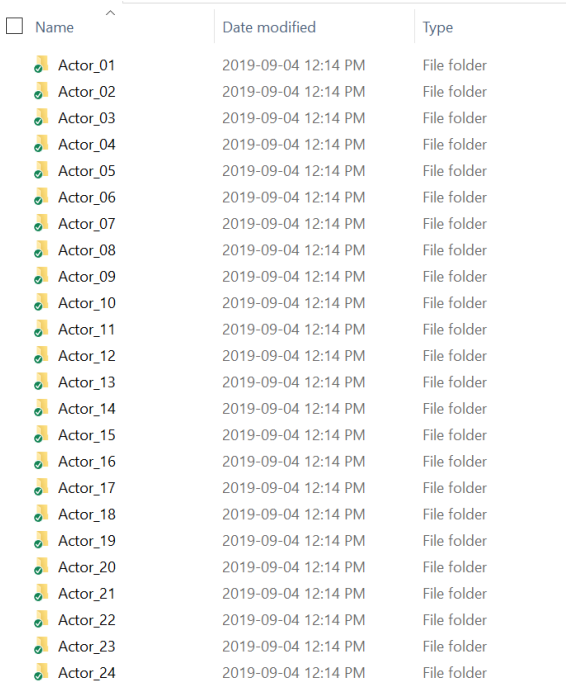

Figure 3.2. Overall view of the RAVDESS dataset

Audio data includes 1440 files, 16bit, 48kHz in .wav format performed by 24 professional actors (12 females and 12 males). Each actor or actress participates in 60 trials. The language is English spoken with a North American accent. The actors vocalize two sentences which by nature are sentiment-free and neutral. The two statements are: "Kids are talking by the door", and "Dogs are sitting by the door". The emotions in this dataset are seven primary emotions: Angry, Calm, Disgusted, Fearful,

Happy, Sad, Surprised in addition to a Neutral state. The emotions are demonstrated with two level of intensity: once normally and once, strongly. However, for neutral state there is no strong intensity.

In general, naming of the files contains seven numbers as identifiers that indicates the characteristics of each file. The coding system of identifiers is explained below.

- Modality (01 = Audio-Visual, 02 = Video-only, 03 = Audio-only)
- Vocal Channel (01 = Speech, 02 = Song)
- Emotion (01 = Neutral, 02 = Calm, 03 = Happy, 04 = Sad, 05 = Angry, 06 = Fearful, 07 = Disgust, 08 = Surprised)
- Emotional intensity (01 = Normal, 02 = Strong)
- Statement (01 = “Kids are talking by the door”, 02 = “Dogs are sitting by the door”)
- Repetition (01 = 1st repetition, 02 = 2nd repetition)
- Actor (01 to 24)

All the files used for this study starts with code 03 and continues with code 01 as it is a speech emotion recognition study. Also, note that odd numbers in terms of actors’ number are assigned to males and even numbered actors are females. An example is presented by Figure 3.3.

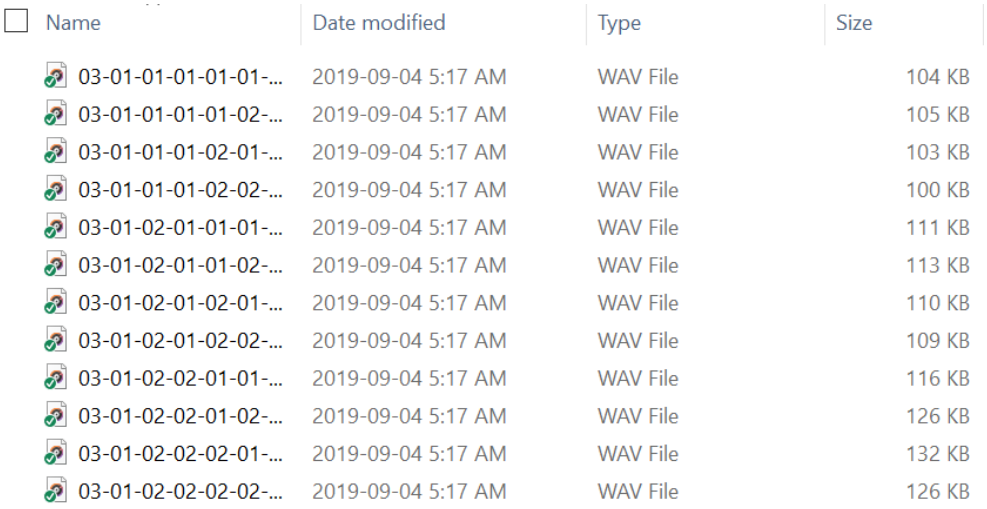

Figure 3.3. an example of naming of the files

The reasons behind this selection among other available options are to fulfill several requirements. First, as mentioned at the beginning of this section, the dataset is not only publicly available, but also free of charge. Although it is labeled in codes, a defined dictionary in python code could easily interpret the code into labels. In conclusion, the data is already labeled even though in codes. Moreover, the dataset

covers a sufficient variety of emotions in English language. More importantly, it contains “calm”, “sad”, and “neutral state” which helps answering the question of how distinguishable from each other these three emotions are. In the end, on the contrary of another famous and suitable dataset, Toronto emotional speech set (TESS) [12], RAVDESS includes both male and female voices. Although it is possible that the process of the training would be more challenging with a lower accuracy, but the result is more inclusive.

Since the dataset is coded, I used a dictionary in python in order to clarify the labels associated with each code. The dictionary is presented in Figure 3.4.

```
32    emotions = {
33        '01': 'neutral',
34        '02': 'calm',
35        '03': 'happy',
36        '04': 'sad',
37        '05': 'angry',
38        '06': 'fearful',
39        '07': 'disgust',
40        '08': 'surprised'
41    }
42
```

Figure 3.4. Defined Dictionary for Labels

The numbers in the name of each file are split with dashes (-) between them (e.g., 03-01-01-01-01-01-02) and as previously mentioned, the third number indicates the type of emotion, hence, the main label of data. In order to acquire the label after the dictionary is defined, one line of code suffices. The code is presented in Figure 3.5.

```
46    def load_data(test_size=0.2):
47        x, y = [], []
48        for file in glob.glob(
49                "C:/Users/armin/Box/Documents/University Papers/"
50                "Thesis/Dataset/Ryerson/RVDESS/Actor_*/*.wav"):
51            file_name = os.path.basename(file)
52            emotion = emotions[file_name.split("-")[2]]
53            if emotion not in observed_emotions:
54                continue
```

Figure 3.5. Loading data and Pin-pointing the Labels

Turning the codes into strings, the third space (excluding the dashes) is occupied by the required label. As the counting in strings start from zero, the number associated with the third room would be 2 which explains the number in the code.

3.3. Feature Extraction

The next step before generating the model and after picking a proper database for training an algorithm is features extraction. In this study, three features are considered for the MLP classifier. Trying to implement more features, especially the ones requiring heavy computation, the traditional classifier loses its most important advantage. This advantage is mostly the fact that despite the lower accuracy, comparing to more advanced algorithms, traditional algorithms perform faster and require less processing power. However, it does not limit the choices for the sake of experiment. As a result, alongside two the most favorable features in Speech Emotion Recognition field, Mel-frequency cepstral coefficients (MFCC) and Mel frequency cepstrum (MFC), a third feature referred to as Chroma which is condensed form of a tonal content of a musical signal is utilized. The latter is comparatively new in this field; only in order to examine the performance of model with a new feature.

3.3.1. Mel spectrogram

To explain the concept of Mel Spectrogram, first, I would go through spectrogram of a signal, specifically, a voice signal.

Since a voice signal in general is a sequence of vibrations which carry energy, in order to have a spectrogram for studying the signal, visualizing the airwaves is the key. A simple visualization would be possible by the brief coding presented in Figure 3.6.

```
import librosa
import librosa.display
import numpy as np
import matplotlib.pyplot as plt
y, sr = librosa.load('C:/Users/armin/Box/Documents/University Papers/'
                     'Thesis/Dataset/Ryerson/Actor_1/Strong Angry 1.wav', sr=32000, mono=True)
Angry, _ = librosa.effects.trim(y)
librosa.display.waveplot(Angry, sr=sr)
plt.show()
```

Figure 3.6. Simple Visualization Code

However, the result would be simple as well. A two-dimensional picture as shown in Figure 3.7 is not any assistance to the analysis of the wave, even though Librosa library in code above make it considerably easier to visualize the signal.

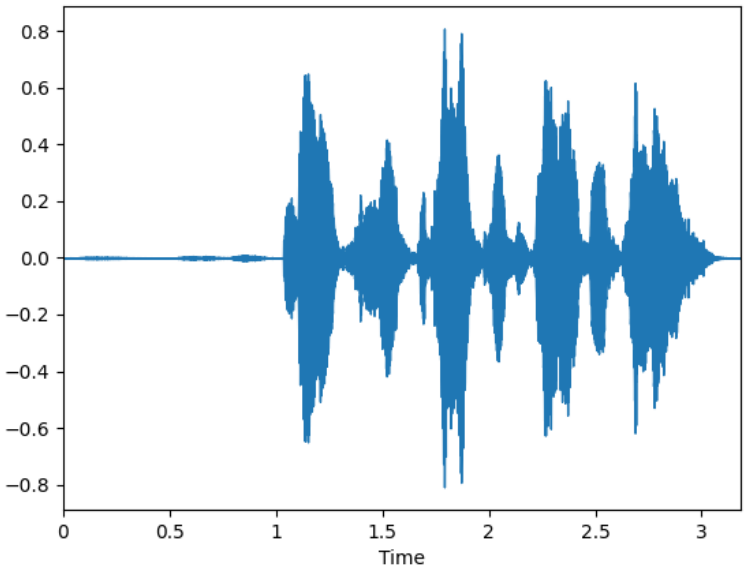

Figure 3.7. Two-dimensional Signal Visualization by the code presented in Figure 3.6

A better example of a spectrogram and the data it could convey is easy to obtain through the code shown in Figure 3.8.

```
import librosa
import librosa.display
import numpy as np
import matplotlib.pyplot as plt
y, sr = librosa.load('C:/Users/armin/Box/Documents/University Papers/'
                     'Thesis/Dataset/Ryerson/Actor_2/Neutral 1.wav', sr=32000, mono=True)
melspec = librosa.feature.melspectrogram(y, sr=sr, n_mels = 128)
melspec = librosa.power_to_db(melspec).astype(np.float32)
librosa.display.specshow(melspec, x_axis="time",  y_axis="mel", sr=sr, fmax=16000)
plt.show()
```

Figure 3.8. Mel-spectrogram Visualization code

It is a simple code only to understand the concept of spectrogram visualization as shown in Figure 3.9. The more complicated a spectrogram, the more information it carries. Which is why one might apply Fourier Transformation on a signal in order to have frequencies through time and also amplitude, by having a colored spectrogram. The reason I go through these codes and the spectrogram they present is to indicate how Mel-spectrogram works.

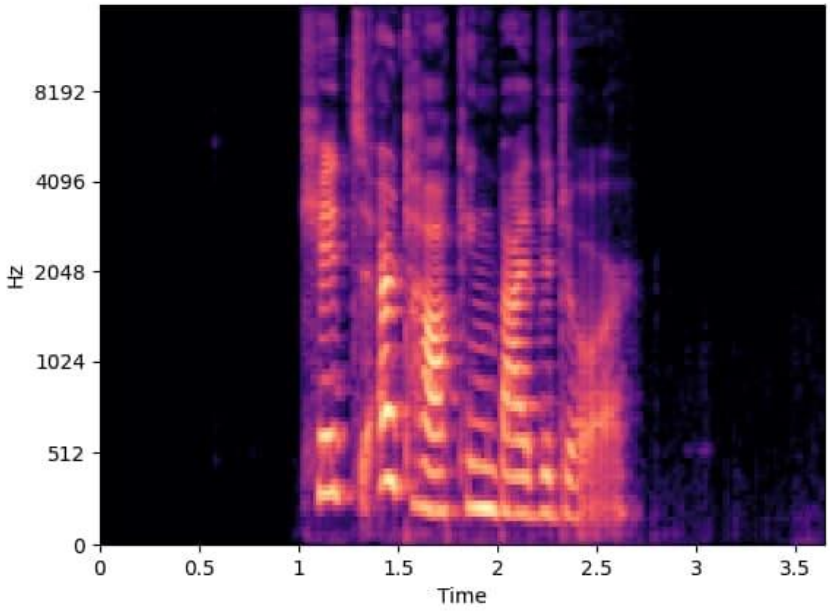

Figure 3.9. Mel-spectrogram Visualization by the code presented in Figure 3.8

Before introducing mel spectrogram, its importance, coding, and results, first a brief introduction of mel scale is required. Mel (after the word “melody” [13]) scale is

technically a perceptual scale which could be through the transformation of a frequency into mel scale. The scale is focused on pitches and works by association of a 1000 Hz voice to a 1000 mels. The formula for the transformation is as follows:

$$\text{mel} = 2595 \log_{10}(1 + \frac{f}{700}) \qquad (3.1)$$

Following the concept of “mel scale”, a mel-spectrogram is a spectrogram with mel scale as one of its axes which usually is y axis. An illustration of the process during which the mel-spectrogram is extracted is presented in Figure 3.10.

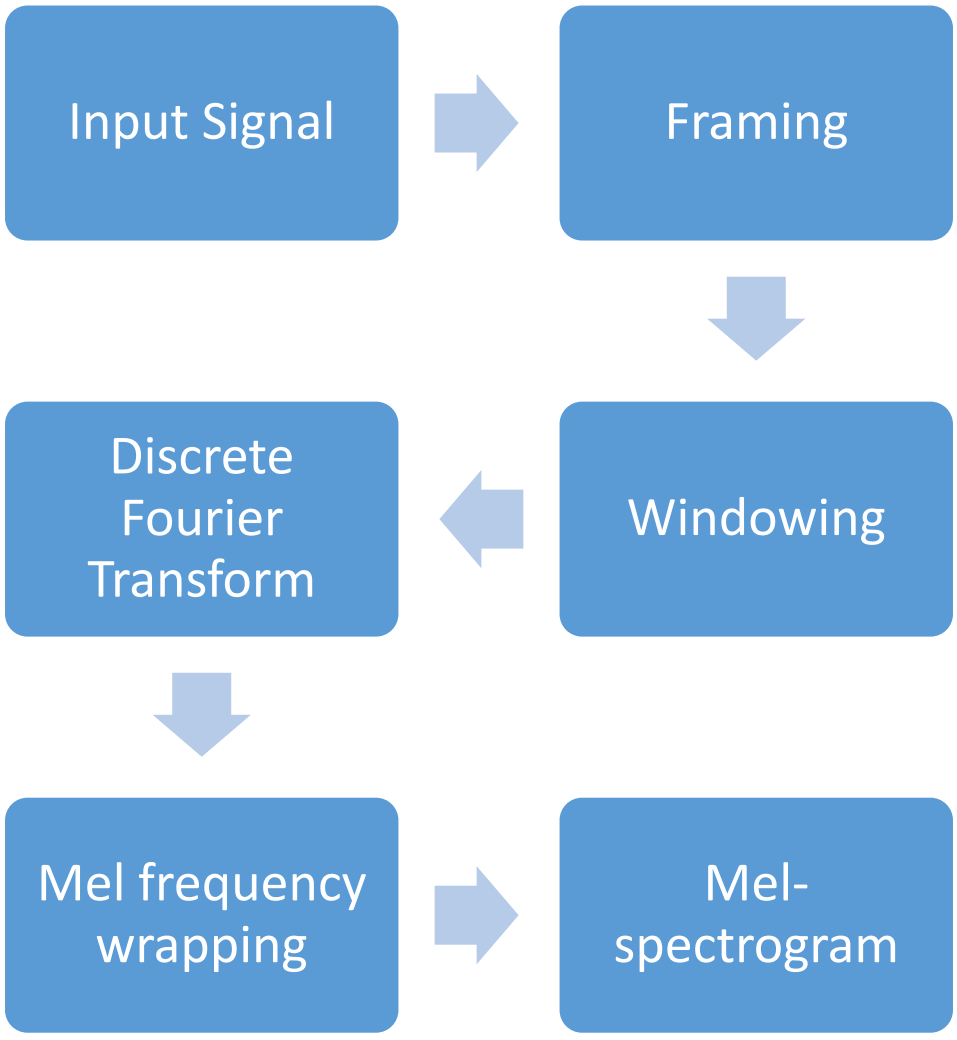

Figure 3.10. A general view of Mel-spectrogram feature extraction process

In cases which are focused on remodeling human hearing or the way a human ear perceives the frequency of a sound (e.g., in parts of emotion recognition by voice,

the effort is to remodel human hearing, the way a human would perceive the voice of another human and determine their feelings), a mel spectrogram which is based on human listener's perceptions is a suitable feature. Additionally, this is why a linear spectrogram is not a favourable spectrogram in this case, analysing the speech signals on which I am working. A linear spectrogram belongs to studies in which all frequencies matter the same.

Fortunately, since I am using Librosa library, it calculates the mel scale automatically and extracts mel spectrogram without requiring mathematical computation from the programmer. The simple code is presented in Figure 3.11.

```
27          if mel:
28              mel = np.mean(librosa.feature.melspectrogram(X, sr=sample_rate).T, axis=0)
29              result = np.hstack((result, mel))
```

Figure 3.11. Librosa library extracting Mel-spectrogram feature

Once an input signal in time domain is provided for Librosa library, it computes the spectrogram of the signal's magnitude and does the mapping onto the mel scale, afterwards. The library returns the Mel-spectrogram feature in the end.

3.3.2. MFCC

In most applications, Mel Frequency Cepstral Coefficients (MFCCs) are the most common features. The fact is due to the MFCC feature's performances which is like the role of human ear, an emotion recognition device that performs better than machine in recognising small changes in voice.

Even though it is mentioned that MFCC is inspired by human hearing, there is an obvious disadvantage with human ears which MFCC needs to cover. When the gap between two different frequencies closes, humans find it difficult to distinguish them. This becomes a bigger issue at high frequencies. Hence, there are Mel filterbanks from narrow to wide for deciding how much energy exists in near zero Hertz to higher frequencies.

A voice signal, or most of signals in general, changes constantly through time. That is why MFCC starts with dividing the signal into short frames so that the emotions

implemented in the signals would be more distinguishable. It is common to divide the signal into frames of 20-40ms length in order to both avoid the constant change and have long enough samples to achieve a reliable spectrum-related analysis. The analysis would be possible by determining the level of energy associated with the frequencies present in that small frame, therefore, the length of samples matters.

To calculate the said amount of power that each frames carries, the periodogram estimate of the frame's spectrum must be calculated. The formula in general is as follows:

$$P_i(K) = \frac{1}{N}|S_i(K)|^2 \qquad (3.2)$$

Where $S_i(K)$ is the Discrete Fourier Transform of the frame and could be computed by the followed equation:

$$S_i(K) = \sum_{n=1}^{N} t_i(n)h(n)e^{-j2\pi kn/N} \qquad 1 \leq k \leq K \qquad (3.3)$$

In this equation, t(n) is the signal which once framed, changes into $t_i(n)$ where i is the number of the frame and n is the number of samples. Moreover, $h(n)$ is the analysis window and K is the length of which one desires to perform the Discrete Fourier Transform.

This is how the periodogram estimate of a power spectrum is obtained. After deciding the amount of energy in the frame, the logarithm of its sum is calculated. There are two reasons behind the choice of logarithm calculation. First, there is another inspiration by human ear, as humans do not perceive voice signals linearly. In other words, to hear a voice two times louder, the energy level is supposed to be eight times higher. The second reason is that this type of operation makes it possible to use a channel normalization method by the name of cepstral mean subtraction.

A schematic of the process explained above is presented in Figure 3.12.

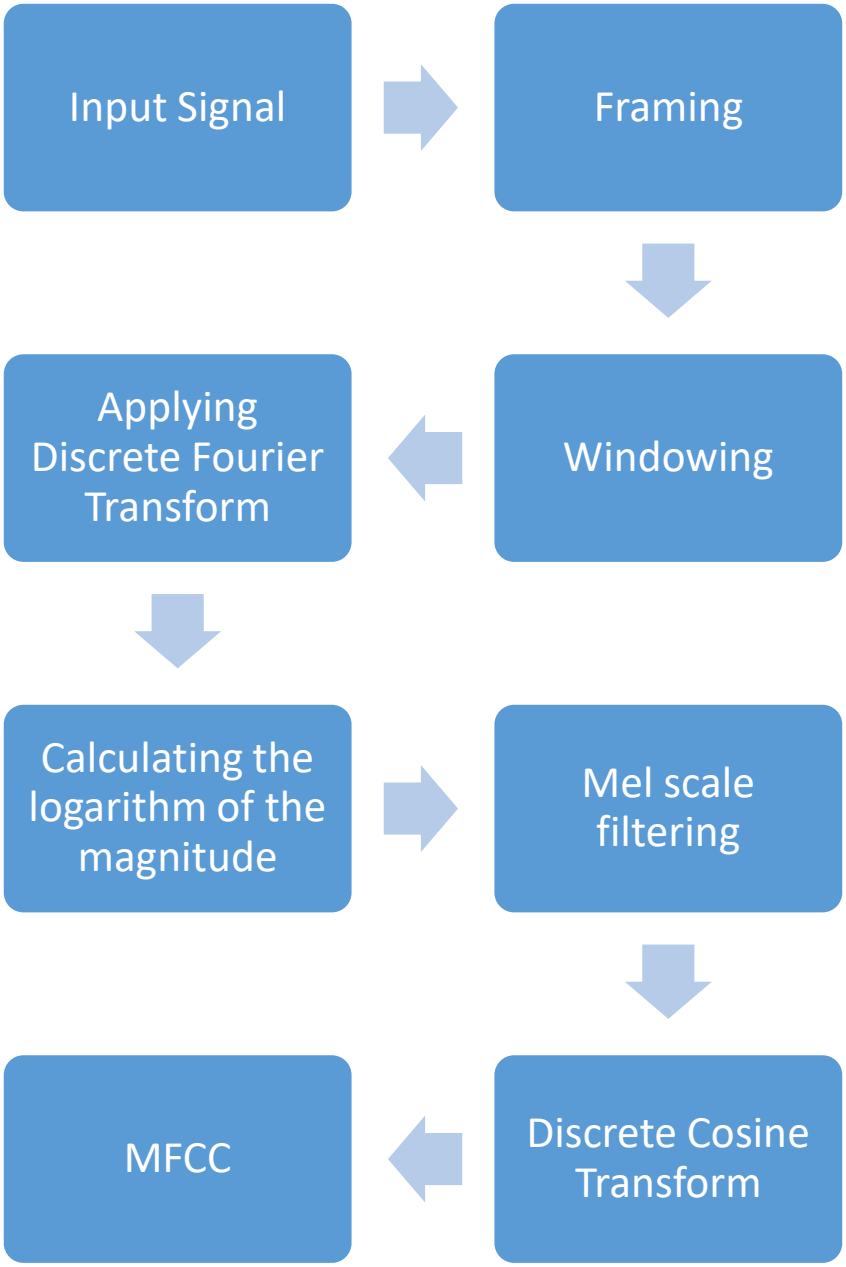

Figure 3.12. A block diagram of a MFCC extraction process

However, while using the Librosa library, there is no need to perform any of these calculations personally. Librosa is capable of calling and extracting features by its own without the need of further computations or further manual manipulations.

The computation steps are widely explained through chapter two of the thesis, in 2.5, even though the steps are not the same in this algorithm as I am using the Librosa library of python. The calculation is automatically done by this library that extracts

features in Speech Emotion Recognition tasks. The procedure is presented in Figure 3.13. It is to mention that Librosa library while going through all the necessary steps, applies STFT as [14] provides proof that FFT results of a sound wave in several cases does not strike the peaks of a signal, which is usually required.

```
21          if mfcc:
22              mfccs = np.mean(librosa.feature.mfcc(y=X, sr=sample_rate, n_mfcc=40).T, axis=0)
23              result = np.hstack((result, mfccs))
```

Figure 3.13. Librosa library extracting MFCC

3.3.3. Chroma

Like MFCC features, chroma features perform in small frames of time. However, they work on magnitude spectrum capturing the pitch-related characteristics of the voice signal. That is why they are also known as “Pitch Class Profile” or PCP. A Chroma vector is (usually) a combination of 12 elements representing different classes of pitch. The vector determines the amount of energy for each of this pitch classes in the signal [15]. This feature which was introduced in [16] is most appropriate when there is an intention to categorize the pitches of a voice signal.

The entire spectrum of a voice signal is divided into 12 classes as mentioned above based on the musical octaves to determine the pitch category of the signal. In music, notes which are exactly one octave apart from each other, would be considered the same. Therefore, a chroma feature is able to categorize the same harmony even if the signal is pitch-wise slightly different. In general, the feature provides model with tonal data on an input signal. A demonstration of this process is presented in Figure 3.14.

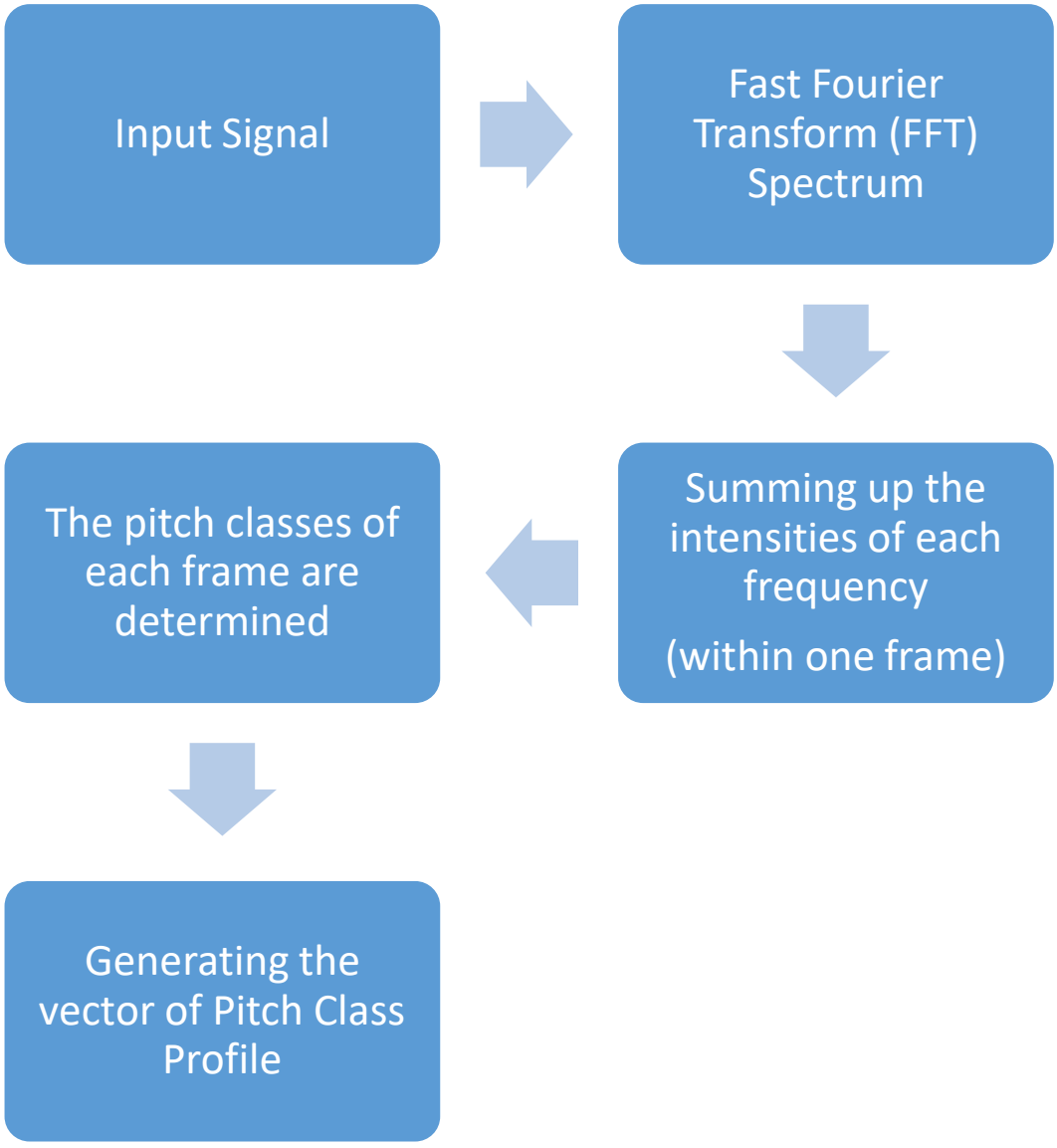

Figure 3.14. A block diagram of a Chroma feature extraction process

Chroma features have been extensively used in analysing music as they capture the harmonic characteristics of the signal regardless of the variation in instruments or the timbers in music. However, it is not a common feature for Speech Emotion Recognition which is why in this model is used to study its effects and usefulness for Emotion Recognition task. I believe Chroma feature increases the accuracy in emotion recognition by voice. The dataset I am using contain both female and male signals.

Those signals even while conveying the same emotion, are remarkably different for a machine when analysing pitches and notes. Hence, a Chroma feature covers for the errors might occur if using only MFCC and Mel spectrogram.

The normal procedure for capturing this feature follows the same principals as the other two. Each voice signal considered as the input acquired a spectrogram which later would be processed, categorized into twelve bins. Normalization and regularization are necessary as well, to perceive the resulted spectrogram.

However, once again using the Librosa library prevent prolonging the process and saves time, energy, and space. A simple command as presented in Figure 3.15 extract chroma spectrogram and feature for the emotion recognition task.

```
24          if chroma:
25              chroma = np.mean(librosa.feature.chroma_stft(S=stft, sr=sample_rate).T, axis=0)
26              result = np.hstack((result, chroma))
```

Figure 3.15. Librosa library extracting Chroma

The library above applies short-time Fourier Transform on the signals (STFT) as the frames result in the shorter signals and the frequency resolution would be higher if instead of FFT, STFT is utilized. Then the process presented in Figure 3.14 results in the PCP vector of the input signal. The vector is part of the numpy array which is the input for the MLP classifier.

3.4. Algorithm

The algorithm that I chose to study a traditional machine learning method on Speech Emotion Recognition is a feedforward Artificial Neural Network, alongside a supervised approach. For this task, I pick a Multi-Layer Perceptron classifier with a single hidden layer in order to distinguish the performances between a Deep Neural Network and a simple feedforward network, more vividly.

The general concept of a MLP classifier, categories, and sub-categories have been previously mentioned. In this section, I am planning to explain the code line by line and justify the choices I made throughout the process. The classifier, just as many others, has elements that one would choose between them according to their needs and

tasks. However, given the fact that this task is not a peculiar task and mostly considered as simple signal processing, the main focus of this part is on feature selection and the dataset, not on the algorithm itself.

Before starting to explain about the body of MLP classifier, Figure 3.16 presents the function defined in order to load the dataset, extract features and labels and return them by the time it is called.

```
46    def load_data(test_size=0.2):
47        x, y = [], []
48        for file in glob.glob(
49                "C:/Users/armin/Box/Documents/University Papers/"
50                "Thesis/Dataset/Ryerson/RVDESS/Actor_*/*.wav"):
51            file_name = os.path.basename(file)
52            emotion = emotions[file_name.split("-")[2]]
53            if emotion not in observed_emotions:
54                continue
55            feature = extract_feature(file, mfcc=True, chroma=True, mel=True)
56            x.append(feature)
57            y.append(emotion)
58        return train_test_split(np.array(x), y, test_size=test_size, random_state=9)
```

Figure 3.16. Function for loading the dataset and extracting features and labels

Afterwards, the code continues to split the training set and the test set, defining the classifier, training the model, and in the end, calculating its accuracy. The code alongside the comments for a better understanding is shown in Figure 3.16.

```
61    # Split the dataset
62    x_train, x_test, y_train, y_test = load_data(test_size=0.25)
63    print((x_train.shape[0], x_test.shape[0]))
64    print(f'Features extracted: {x_train.shape[1]}')
65
66    # Initialize MLP
67    model = MLPClassifier(alpha=0.01, hidden_layer_sizes=(300,),
68                          solver='lbfgs', learning_rate='adaptive',
69                          max_iter=700)
70    # Train
71    model.fit(x_train, y_train)
72    # Predict
73    y_pred = model.predict(x_test)
74    # Calculate the accuracy
75    accuracy = accuracy_score(y_true=y_test, y_pred=y_pred)
76
77    # Print the accuracy
78    print("Accuracy: {:.2f}%".format(accuracy * 100))
```

Figure 3.17. MLP classifier code and the comments

The ratio of training set to test set is usually 1:3. In other words, train set normally consists of 25% of the whole data and the 75% of it is saved for testing the model. I slightly changed the ratio and increased the percentage of training set to 30% which did not affect the accuracy. Hence, I decided on keeping the normal ratio.

After acquiring the shape of training and dataset and determining the number of extracted features, in line 67 to 69 the MLP classifier is implemented. The default activation function is the Rectifier Linear Unit also known as ReLU. Since it covers most of the numerical problems of other activation functions, I do not define any replacement for this default adjustment. However, as the default solver for MLP classifier, Adaptive Linear Momentum (adam), is usually used for big data analysis [17] and the dataset for this study is relatively small, I changed the solver to solver to the Broyden, Fletcher, Goldfarb, and Shanno, or BFGS Algorithm from the family of Quasi-Newton methods which suits small datasets better [17]. As a result, some other elements are eliminated since they only work with adam solver or LBFGS does not used them. For example, having LBFGS as a solver, the model does not use minibatch as LBFGS suggests that the dataset is already too small to utilize batches.

The change of learning rate from default “constant” to “adaptive” is a personal preference. Also, for determining the number of maximum iterations, I started from the default and as long as the accuracy increased, I increased the number. 700 of iterations provided the model with the highest accuracy and when I change the number to 800, the accuracy drops. That is why I settled on the number of 700 for maximum iterations.

In conclusion, the procedure of this algorithm is as follows: First, it is provided with the dataset which in case is already denoised and ready for application. Second, the aforementioned features are extracted by Librosa library and settled into a numpy array for each signal, alongside the labels. These arrays shape the input for MLP classifier that starts training on them, afterwards. The training method for MLP classifier is Backpropagation, short for “backward propagation of errors” [18]. This training algorithm is utilized in supervised-learning cases for Neural Networks, using gradient descent. After training the network with the train set determined for the task, the test set would be going through the classifier. Their classes would be predicted and are the final output of both classifier and the algorithm in general.

The flowchart for the algorithm is presented in Figure 3.18. This is the final view of the classifier I implement on the Ryerson dataset. The next step is running the algorithm and delivering the results.

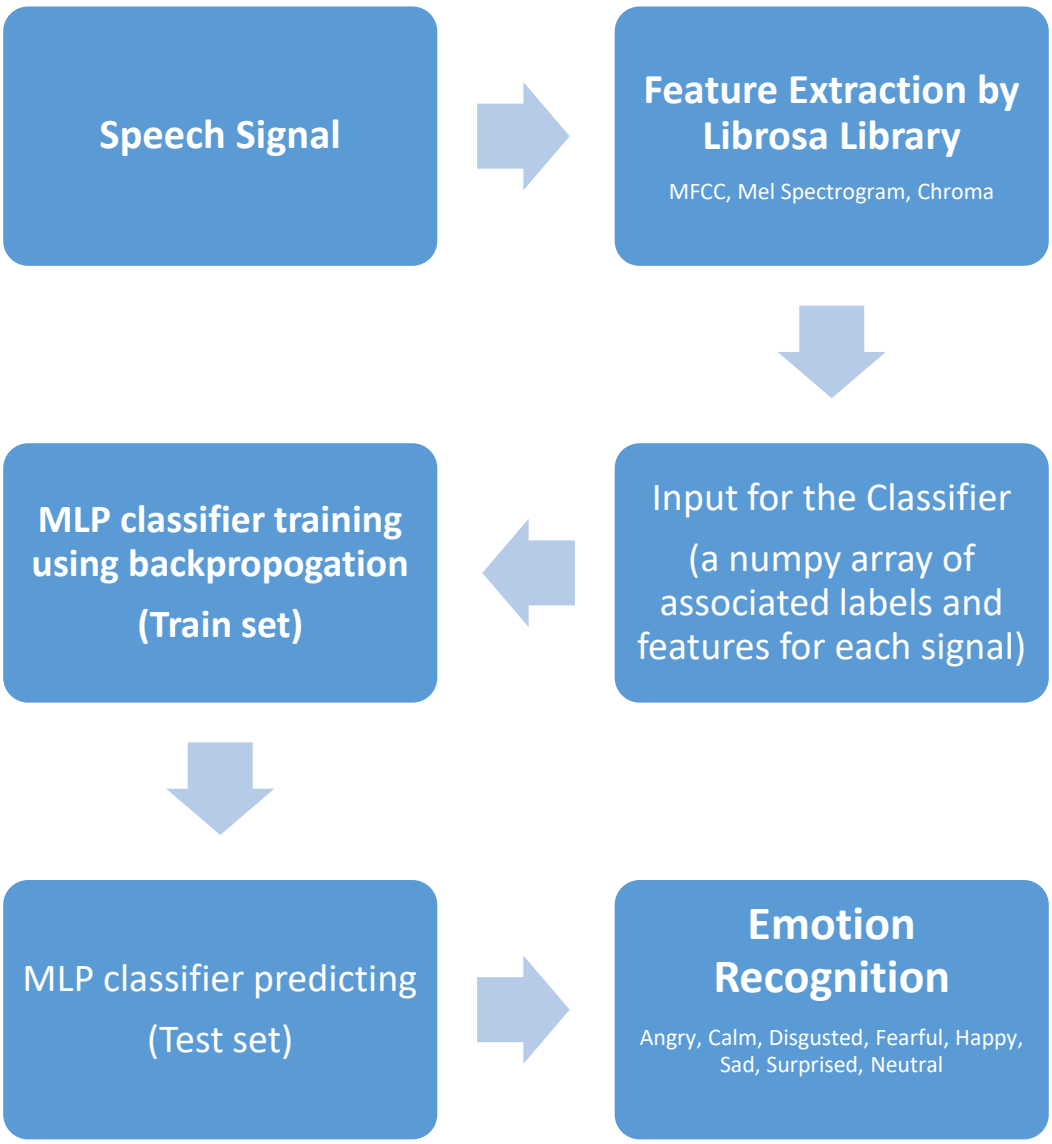

Figure 3.18. The general view of the method used in this chapter

3.5. Results

By the end of my algorithm, I also add a simple code to generate a confusion matrix after the training and a better understanding of the performance of MLP classifier. The code is presented in Figure 3.19.

```
82    cm = confusion_matrix(y_test, y_pred)
83    index = ['angry', 'calm', 'disgust', 'fearful', 'happy', 'neutral', 'sad', 'surprised']
84    columns = ['angry', 'calm', 'disgust', 'fearful', 'happy', 'neutral', 'sad', 'surprised']
85    cm_df = pd.DataFrame(cm, index, columns)
86    plt.figure(figsize=(10, 6))
87    sns.heatmap(cm_df, annot=True)
88    plt.show()
```

Figure 3.19. Confusion Matrix code

The initial result for the code is 42.38%, recognising all seven primary emotions. Limiting the emotion to four (Calm, Happy, Fearful, and Disgust which are the most different) increased the accuracy to 69.48%. The model consisted of three features which in each try, I eliminate one to observe the difference in performance. The accuracies without Mel spectrogram, MFCC, and Chroma all drop to 40.83%, 37.50%, and 40.83% respectively. The numbers prove the importance of MFCC which is already known as Most Frequently Considered Coefficients among the experts. It appears that mel and chroma spectrograms are exactly as efficient as one another with respect to increasing the accuracy.

The confusion matrix for the main code is shown in Figure 3.20 that explains why Sad and Neutral increases the accuracy considerably. The three emotions, Sad, Calm, and Neutral are mostly confused with each other.

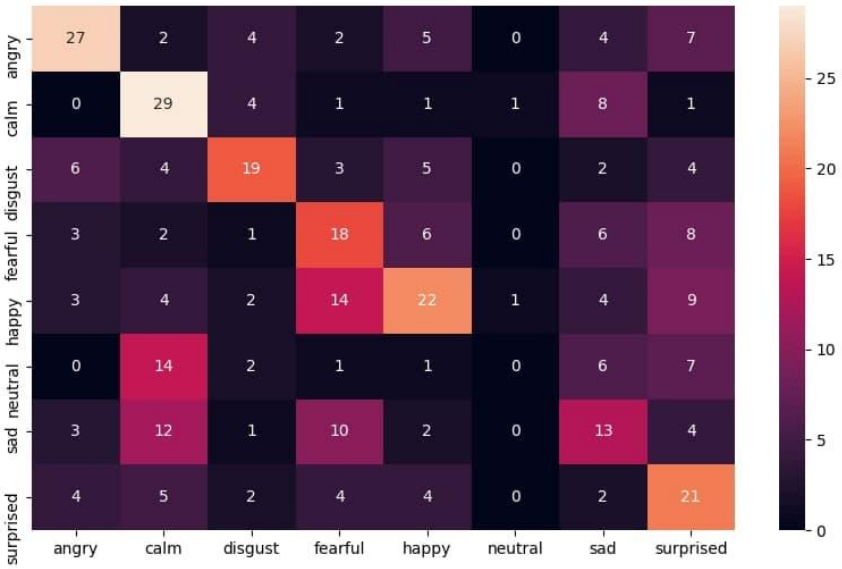

Figure 3.20. Confusion Matrix of the MLP classifier (Rows represent correct labels; columns are predicted labels.)

The confusion matrices for classifier without MFCC, mel and chroma spectrogram is shown below. In Figure 3.21 it is demonstrated that once MFCC is removed from the algorithm, other than Calm and Neutral which previously were confused for one another, the accuracy for detection of all the other emotions decreases.

Figure 3.21. MLP classifier without MFCC

The confusion matrix presented in Figure 3.22 delivers another interesting result: By eliminating the mel-spectrogram, the confusion between Anger, Happy and Surprised voice signals which all tend to have high pitches similar to one another decrease. However, the final accuracy is still lower than an algorithm with mel-spectrogram as the confusion for other emotions increase.

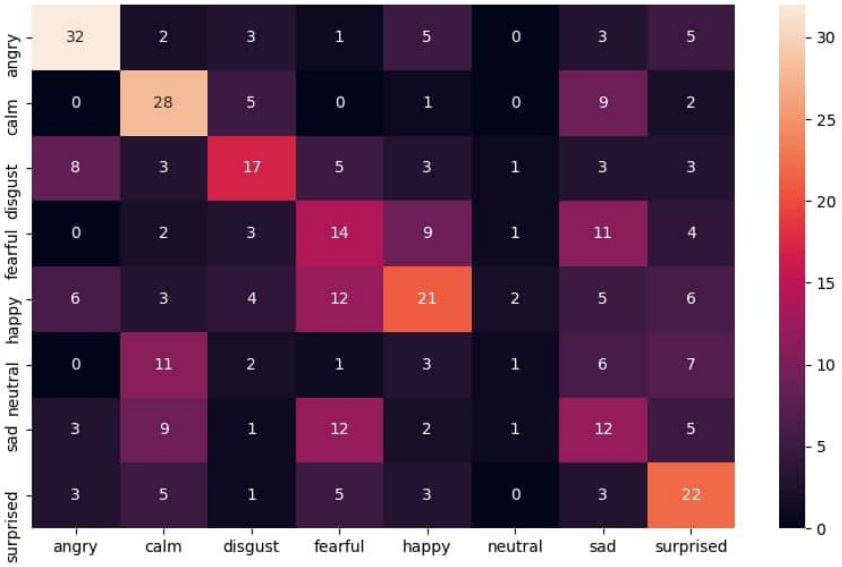

Figure 3.22. MLP Classifier without Mel-spectrogram

Figure 3.23 shows that having the Chroma feature removed appears to have a small impact on all the emotions, except for the detection of neutral emotional state which improves remarkably. As mentioned before, chroma classifies the harmony in the audio files and categorize them into 12 pitch classes. The harmony in a neutral voice is possible to be mistaken with Calm and Sad, even for a human ear. Hence, once there is no categorizing by the harmony, Neutral state is easier to detect, even though the same statement is not true for the rest of emotions.

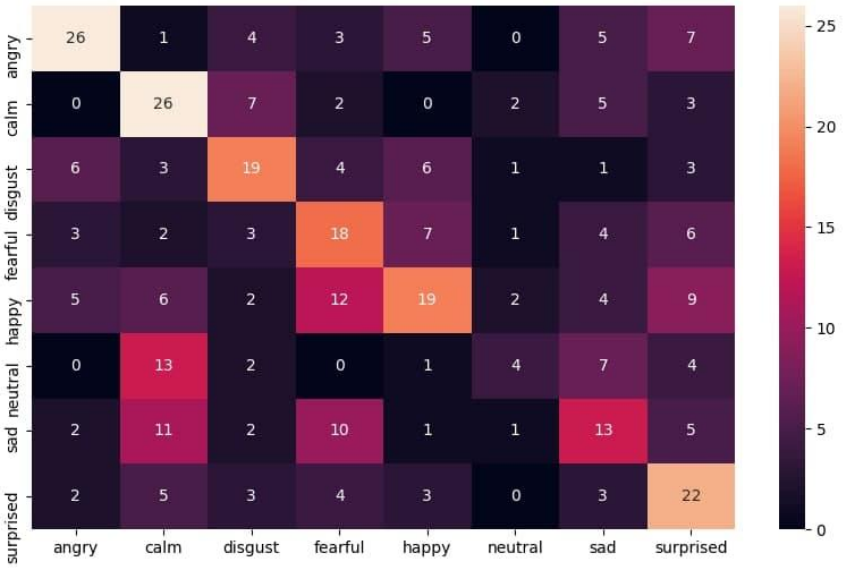

Figure 3.23. MLP classifier without Chroma

3.6. Conclusion

In short, neither the combination of three features, MFCC, Mel spectrogram, and Chroma could fully overcome the limitations of MLP classifier without more than one hidden layer, nor any other combination between the features. The classifier delivers poor results in general, not only in comparison to a Deep Neural Network, but also comparing to other traditional methods. All the results for this classifier is presented in Figure 3.24. Neither of bars showing accuracies for the model with various combinations of features reaches at least 50% of accuracy.

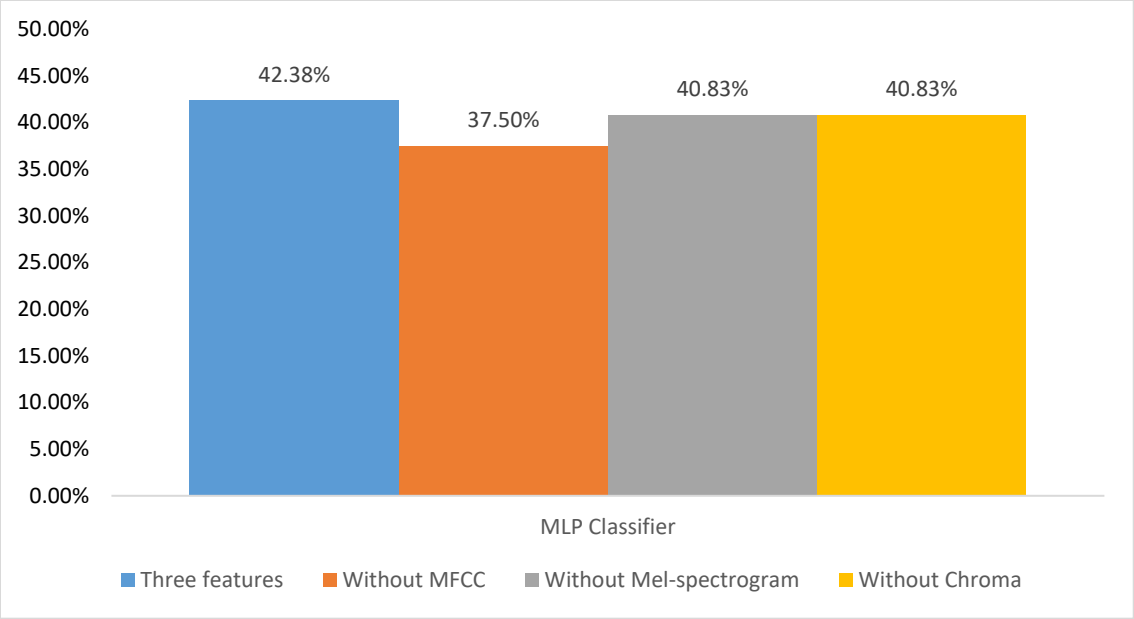

Figure 3.24 Accuracies for MLP classifier

However, even though categorising all the seven emotions appeared to be a challenging task for a MLP classifier and decreasing the number, specifically not having Neutral, Calm, and Sad emotions at the same time, improves the accuracy, features proved themselves effective and suitable. As the three features cover different areas and approaches, I believe for better results in terms of accuracy and preventing the confusion between similar emotions a speech processing would be more helpful than adding other signal processing features. It is possible to achieve better results through implementing other traditional methods such as SVM, GMM, or HMM in the hidden layers of MLP classifier and utilizing the proposed features at the same time.

Chapter 4.

Emotion Recognition through Voice using Deep Sequential Neural Network

4.1. Introduction

While features such as MFCC and Mel-spectrogram try to mimic human hearing in order to achieve higher accuracies in emotion recognition field, deep learning has been an emerging solution to most of Human-Computer Interactions (HCI). It is utilized to imitate the performance of human brain to deliver better understanding in the sense of interacting with humans [1]. Deep learning methods are more complex in comparison to traditional machine learning methods as they are composed of several non-linear layers computing in parallel. However, as Sidorov et al. provides a comparison between several traditional methods and deep learning techniques, deep networks in general outperform traditional machine learning methods [2]. In [2] a KNN, SVM, and Deep CNN algorithm is applied on IEMOCAP, Surrey Audio-Visual Expressed Emotion (SAVEE), and Emo-DB. The Table of comparison was presented in detail in chapter 2 of this thesis, Table 2.7, but in short, DCNN outperforms all traditional techniques in recognising Angry, Happy, and Sad emotion.

There are many developed algorithms presented throughout the years of dedicated works towards Speech Emotion Recognition by deep learning techniques such as [3], [4], [5], and [6]. A few of these works are listed in Table 4.1. There are also papers such as [7] that presents comparison in terms of accuracy between a CNN (87.74%), a LTSM (79.87%), and a distributed CNN (88.01%) once applied on Berlin Database of Emotional Speech (Emo-DB) [8]. In another study, [9] provided a repost of the accuracies of various RNN structures.

Table 4.1. few papers on Deep Learning techniques for SER

Author (reference)	Emotions	Dataset	Notes	Accuracy

Dengke Tang et al. [10]	Anger, Happiness, Neutral, Sadness	EmotAsS dataset	A combination of CNN (feature extraction) and RNN (feature analysis), classified by softmax function.	45.12% (on the balanced dataset)
Mousmita Sarma et al. [11]	Anger, Disgust, Fear, Happiness, Sadness, Surprised, Neutral	IEMOCAP dataset	The effect of different features extraction methods, various design of layers and time pooling, chunk lengths and epochs in a TDNN-LTSM model is studied.	Respective accuracies for each comparison presented in the paper
H. M. Fayek et al. [12]	Anger, Boredom, Disgust, Joy, Sadness, Neutral	eNTERFACE and the Surrey Audio-Visual Expressed Emotion (SAVEE) database	A real-time Speech Emotion Recognition deep learning model with an end-to-end architecture. A DNN is applied and the comparison between eNTERFACE (6 emotions) and SAVEE (7 emotions) is presented.	60.53% (eNTERFACE) 58.7% (SAVEE)

A more elaborate review on complex deep neural networks such as AEs, RNNs, CNNs, and GANs is presented in the literature review section of this thesis. The general approach is to augment those methods through combining them in order to overcome the limits of each. Whereas in this thesis and specifically in this chapter a simple approach is taken towards deep learning to determine the threshold of this path instead of high limits.

In chapter 3, I employed an Artificial Neural Network with a single hidden layer to detect emotions. In order to compare the performance with a deep learning approach, in this chapter, I apply the Sequential Feed-forward Neural Network that is categorized within the Deep Neural Network classification. The number of hidden layers is 4 and the type of all the hidden layers is Dense. In other words, the algorithm does not contain any Convolutional or Recurrent layer to augment the network. This Sequential Network with Dense hidden layers is specifically suitable when there are single inputs and outputs, rather than multiple tensors. Moreover, it is not a non-linear modeling and topology task, nor does require layer-sharing. Hence, Sequential model proves to be more appropriate.

Another reason behind the selection of this method lies behind the study conducted in [13] that observes the pros and cons of Deep Neural Network,

Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) (inclusively, LTSM). The study concludes that by sacrificing performance and increasing the number of elements, a DNN works comparatively faster than CNN and RNN. This is why I prefer to run a Sequential Deep Neural Network instead of Convolutional or Recurrent Neural Network while utilizing a Core i7 CPU.

The Ryerson dataset is previously explained thoroughly and since this DNN method does not require a dictionary for beforehand labeling, the dataset section is not repeated in this part. For the purpose of a better comparison solely between the two models, I also add a feature extraction step and limit the DNN model to the three features explained in Chapter 03. Hence, there is no repetitive review on the features as well, even though the coding process is slightly different. However, there is a study on the performance of this Sequential Deep Neural Network removing each feature. The results are presented by the end of this chapter.

4.2. Algorithm and Approach

The overall approach towards generating a model in neural network is expressed by Chollet [14] and presented in Figure 4.1. In general, the process starts with determining the data and the percentage dedicated to each training and test sets, followed by defining the Neural Network model by its parameter to later go through the learning process that is defined.

The training process and its configuration depend on the algorithm of its optimizer. Defining Neural Networks, there are various options for optimizers which [15] addresses some of the most important ones. Adaptive Linear Momentum (Adam) optimizer is one of which, used in this model [16].

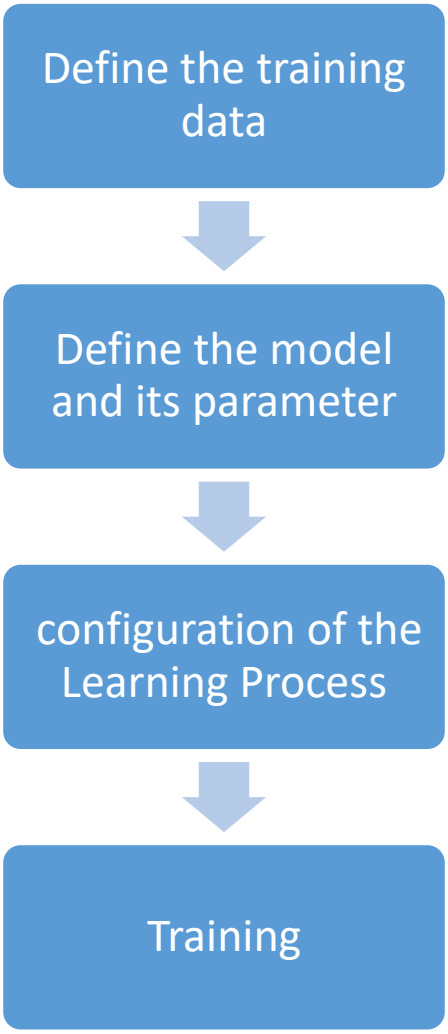

Figure 4.1. Generating a Neural Network model

Before starting to explain the body of model, which is the Deep Neural Network and its parameters, I go through two specific functions I utilize for sorting the input, labels, and feature. Since one of them is specifically considered for the purpose of increasing the accuracy of the model.

First, in order to extract features and labels from raw data, instead of simply going through files by a plain loop order, I use parse_audio_files function which is defined as shown in Figure 4.2.

```
26    def parse_audio_files(parent_dir,sub_dirs,file_ext="*.wav"):
27        features, labels = np.empty((0,180)), np.empty(0)
28        for label, sub_dir in enumerate(sub_dirs):
29            for fn in glob.glob(os.path.join(parent_dir, sub_dir, file_ext)):
30                try:
31                  mfccs, chroma, mel= extract_feature(fn)
32                except Exception as e:
33                  print ("Error encountered while parsing file: ", fn)
34                  continue
35                ext_features = np.hstack([mfccs,chroma,mel])
36                features = np.vstack([features,ext_features])
37                labels = np.append(labels, fn.split('/')[-1].split('-')[2])
38        return np.array(features), np.array(labels, dtype = np.int)
```

Figure 4.2. parse_audio_files function

As Figure 4.2 shows, the function expects three inputs; Parent directory, subdirectory, and the extension of files which in this case, is ".wav". Afterwards, the function delivers two outputs (once again, in this case) which are the labels and features associated with each file. The labels and their corresponding features are saved into a numpy array in order to train the model later on.

After defining the function and before extraction of features and labels, another series of codes is required which is presented in Figure 4.3. This code provides the function with the inputs it demands before calling it and asking for the output.

```
49    main_dir = 'C:/Users/armin/Box/Documents/University Papers' \
50               '/Thesis/Dataset/Ryerson/RVDESS'
51    sub_dir = os.listdir(main_dir)
52    print("\ncollecting features and labels...")
53    print("\nthis will take some time...")
54    features, labels = parse_audio_files(main_dir, sub_dir)
55    print("done")
```

Figure 4.3. Parent Directory and Subdirectory data

However, there is one method left to propose before the start of saving labels and features and feeding the model with them.

As the command categorizes data into different classes, there are both numerical as well as categorical values assigned to each class. There is a problem that this

procedure would cause to the accuracy of the model. The model assumes as the numerical value of an entry increases, the categorical value gets higher. In other words, when there are classes in row one, two, and three of the entries, model supposes the more data is classified into class three (higher number), it is more accurate. It confuses the number of categories with their values. In conclusion, the more data is classified into categories with higher “value”, the higher the accuracy of the model would be, which is wrong.

As a result, I also apply “one hot encoding” method before running into a problem of that sort. One hot encoding turns the categorical values into binary code and regard the categories themselves as feature. In other words, if a file belongs to the class of emotion “anger”, the value for anger “feature” is 1, while all the other categories are considered as 0.

The body of aforementioned function is presented by Figure 4.4.

```
40    def one_hot_encode(labels):
41        n_labels = len(labels)
42        n_unique_labels = len(np.unique(labels))
43        one_hot_encode = np.zeros((n_labels, n_unique_labels+1))
44        one_hot_encode[np.arange(n_labels), labels] = 1
45        one_hot_encode = np.delete(one_hot_encode, 0, axis=1)
46        return one_hot_encode
47
```

Figure 4.4. One hot encoding

Once the processing of raw data is over, the trimmed information is ready to be saved and fed to the model. As mentioned above, features and labels are stored as numpy arrays and go through the loading phase as presented in Figure 4.5. The last line of presented code in below figure is splitting the dataset into test and training set. I started with the usual 25% test size, but out of curiosity tried a higher fraction of data for testing and the accuracy dramatically decreased. Therefore, I change the ratio back to the usual.

```
59    np.save('X', features)
60    # one hot encoding labels
61    labels = one_hot_encode(labels)
62    np.save('y', labels)
63    X = np.load('X.npy')
64    y = np.load('y.npy')
65    train_x, test_x, train_y, test_y = train_test_split\
66        (X, y, test_size=0.25, random_state=42)
```

Figure 4.5. Saving and loading features and labels

As the primary problems are solved, the code goes to determining the parameters of the Deep Neural Network using for this task. The first parameters with which to start are layers and their units. The process is presented in Figure 4.6.

```
67    # network parameters
68    n_dim = train_x.shape[1]
69    n_classes = train_y.shape[1]
70    n_hidden_units_1 = n_dim
71    n_hidden_units_2 = 400  # starts with n_dim * 2
72    n_hidden_units_3 = 200  # half of previous layer
73    n_hidden_units_4 = 100  # half of previous layer
```

Figure 4.6. The layers and their units

It is specifically important to start the process with clarifying the shape of inputs. Shapes are representatives of the number of elements the array (or tensor in other cases) possess in each dimension.

When the Hidden layers receive the input, they process it by applying their weights on it. However, they cannot generate weights and transform input shapes into output shapes unless the shape of input is determined first. Different types of layers require certain number of dimensions to decide the shape of input. Dense hidden layers demand inputs as (batch_size, input_size). After the first round of processing initial

inputs and their shapes, layers start initializing the weights which would be calculated automatically based on the input and output shape.

The general shapes of hidden layers are decided, and they are ready to be utilized in the model. There are other parameters to specify in order to create a Deep Neural Network. The body of code is presented Figure 4.7. It is a sequential network as explained in the introduction, with a ReLU activation function, an adam optimiser [17] and the dropout rate of 20%. The rate is chosen according to the results [18] provided in their study. It appears that this percentage prevent over-fitting problem (the same task assigned to Regularization step in traditional Machine Learning methods) without sacrificing accuracy.

The layers are added incrementally, not all at once, by add () command. They could be easily removed, for observing purposes by pop () command. Even though the activation function for each layer remains the same through the four hidden layers, for the output layer a softmax function is applied as it is the norm. The reason behind this choice in cases where output categories are predicted, is to normalize the final output of the network. The basic formula for this function is as follows:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} 1^{e^{z_j}}} \qquad \text{for i = 1,…,K and z = } (z_1, \ldots, z_k) \in \mathbb{R}^k \qquad (4.1)$$

Where z is the input vector of K categories and $\sigma(z)$ which is the softmax function is supposed to normalize it over K number of probabilities.

```
# the model
def create_model(activation_function='relu', optimiser='adam', dropout_rate=0.2):
    model = Sequential()
    # layer 1
    model.add(Dense(n_hidden_units_1, input_dim=n_dim, activation=activation_function))
    # layer 2
    model.add(Dense(n_hidden_units_2, activation=activation_function))
    model.add(Dropout(dropout_rate))
    # layer 3
    model.add(Dense(n_hidden_units_3, activation=activation_function))
    model.add(Dropout(dropout_rate))
    # layer4
    model.add(Dense(n_hidden_units_4, activation=activation_function))
    model.add(Dropout(dropout_rate))
    # output layer
    model.add(Dense(n_classes, activation='softmax'))
    # model compilation
    model.compile(loss='categorical_crossentropy', optimizer=optimiser, metrics=['accuracy'])
    return model
```

Figure 4.7. Deep Neural Network model I

As mentioned before, the hidden layers are dense layers. Each layer provides input for the next layer as follows:

$$\text{Output} = \text{activation (dot (input, kernel)} + \text{bias}) \qquad (4.2)$$

Where output of the layers is the input for the next, activation function is the one defined (in this case, ReLU), and kernel is the weighted matrix that layer creates. After the model receives the shape of the input, Keras would be able to provide the current layer with an input. Also, bias vector is only created by the layer if use_bias is true [19].

The training process starts before generating the model. The Adam optimiser in this model which is in charge of the training process updates the parameters of training procedure as shown below:

$$New\ Parameters\ =$$

$$Parameters\ -\ (\frac{(Gradient\ Exponential\ Decay)}{\sqrt{(Square\ Gradient\ Exponential\ Decay)}})\ .\ (Learning\ Rate) \qquad (4.3)$$

In details, if g_t is the gradient in time t with having default values for both β_1 and β_2 which are decay rates and close to 1, the decaying average of past (ω_t) and its squared gradients (ϑ_t) would be calculated as follows.

$$\omega_t\ =\ \beta_1\omega_{t-1}\ +\ (1\ -\ \beta_1)g_t \qquad (4.4)$$

$$\vartheta_t\ =\ \beta_1\vartheta_{t-1}\ +\ (1\ -\ \beta_2)g_t^2 \qquad (4.5)$$

Since the authors of Adam noticed these values initially are zero-biased, they proposed a bias corrected estimation for both shown below:

$$\widehat{\omega}_t\ =\ \frac{\omega_t}{1-\beta_1^t} \qquad (4.6)$$

$$\hat{\vartheta}_t\ =\ \frac{\vartheta_t}{1-\beta_2^t} \qquad (4.7)$$

Which results in Adam update rule.

$$\theta_{t+1} = \theta_t - (\frac{\eta}{\sqrt{\hat{\vartheta}_t + \epsilon}})\hat{\omega}_t \qquad (4.8)$$

where $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$ with η as learning rate and θ_t is the model parameter at the moment of t.

It should also be mentioned that the loss function for this model is of cross entropy family. The type of function is common in classification problems where the model is predicting the probability of each category. Speaking of which, the type of cross entropy loss function in this case is the categorical one, since the model is provided with one hot encoded data, binary values of 0s and 1s. In conclusion, categorical cross entropy specifically works with binary values which makes it fit for this algorithm.

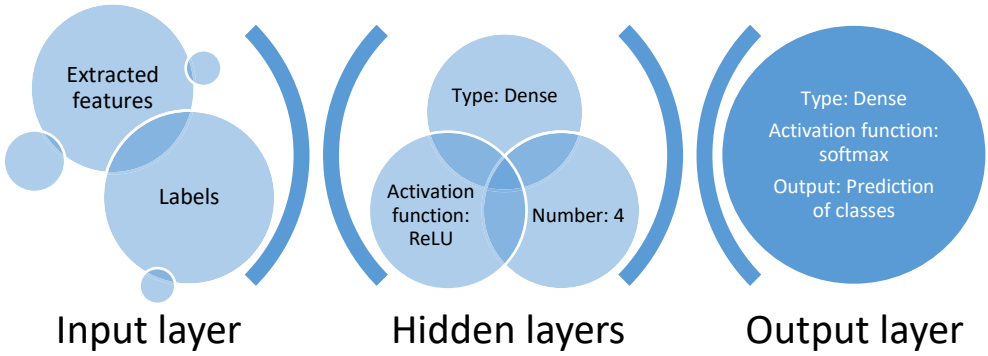

Figure 4.8. Schematics of the Deep Sequential Neural Network

Layers in this case behave similar to a list (of layers). Hence, when the model is to be created, it passes the layers in the list form to the Sequential constructor. This is the function of create_model () command. After that the number of epochs and batch size is determined, model starts predicting and delivering result. By adding a model.summary() command, a summary of the model is asked and provided in Figure 4.9. Additionally, the general view of the network is presented in Figure 4.8.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 180)               32580
_________________________________________________________________
dense_1 (Dense)              (None, 400)               72400
_________________________________________________________________
dropout (Dropout)            (None, 400)               0
_________________________________________________________________
dense_2 (Dense)              (None, 200)               80200
_________________________________________________________________
dropout_1 (Dropout)          (None, 200)               0
_________________________________________________________________
dense_3 (Dense)              (None, 100)               20100
_________________________________________________________________
dropout_2 (Dropout)          (None, 100)               0
_________________________________________________________________
dense_4 (Dense)              (None, 8)                 808
=================================================================
Total params: 206,088
Trainable params: 206,088
Non-trainable params: 0
```

Figure 4.9. The summary of the Deep Sequential Neural Network

In the end, a confusion matrix is applied for a better presentation of performance. The matrices are used for better comparison between the performances of two models, as well.

The rest of the code regarding Deep Neural Network is presented in Figure 4.10. As confusion matrix has been previously studied, it is overlooked in this chapter.

```
    # create
    model = create_model()

    # train
    history = model.fit(train_x, train_y, epochs=200, batch_size=4)

    # predicting from the model
    predict = model.predict(test_x, batch_size=4)
    emotions = ['neutral', 'calm', 'happy', 'sad',
                'angry', 'fearful', 'disgust', 'surprised']
    # predicted emotions from the test set
    y_pred = np.argmax(predict, 1)
    predicted_emo = []
    for i in range(0, test_y.shape[0]):
        emo = emotions[y_pred[i]]
        predicted_emo.append(emo)
        actual_emo = []
        y_true = np.argmax(test_y, 1)
    for i in range(0, test_y.shape[0]):
        emo = emotions[y_true[i]]
        actual_emo.append(emo)
```

Figure 4.10. Deep Neural Network model II

To summarize this section, the process of generating a Deep Neural Network algorithm goes as follows: In case of not having a pre-processed dataset, it starts with denoising and processing the dataset in order to have it ready for the network. Then, in cases such as this study in which features are determined in advance by the author, algorithm starts extracting features from each input signal and gather them alongside the label in a numpy array. The length of this array alternates depending on the extracted features. These arrays shape the input for the Neural Network. As the data is divided into two groups, training set and test set, the training initiates by receiving its input. Note that since there is a Dropout rate determined for each layer, there is a dropout before each hidden layer. Depending on the chosen optimizer while defining the model, the training process might defer. After this phase is complete, the network is ready to be tested so that the accuracy of the generated model is measured. Hence, it starts predicting the labels for its test set. After testing the model, the percentage of its accuracy would be determined and delivered, alongside its confusion matrix.

This process is summarized in Figure 4.11.

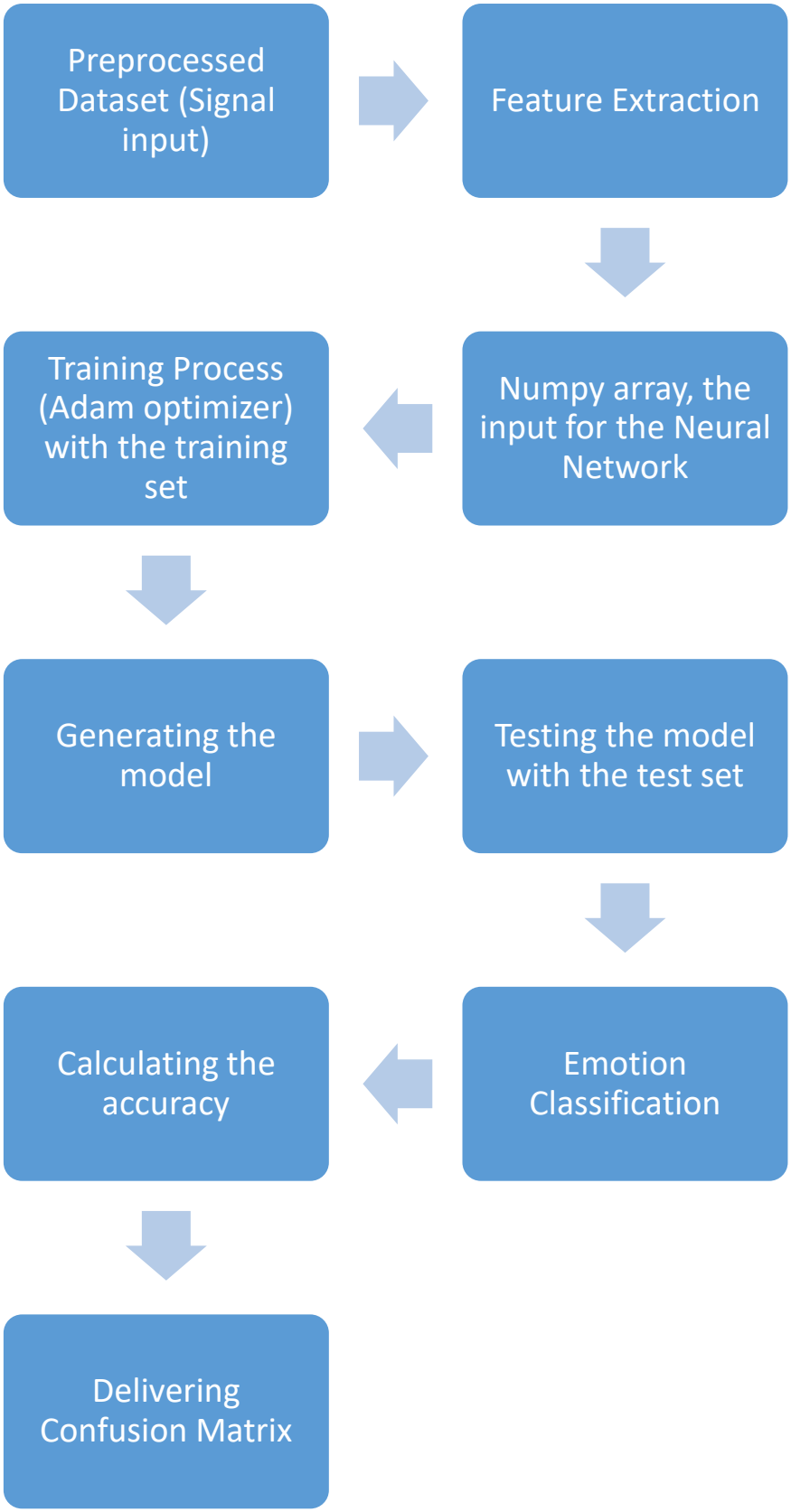

Figure 4.11. The performance of the Deep Neural Network algorithm

4.3. Results

The model is initially run with three features, MFCC, Mel Spectrogram, and Chroma, applied on the Ryerson dataset, resulting 90.57% of accuracy. The confusion matrix for this run of model is presented in Figure 4.12 with actual emotion as rows and predicted categories as columns. As expected, Neutral, Sad, and Calm state are confused the most with one another. The highest accuracy in prediction belongs to Anger and the lowest is Neutral. Angry emotion with its high pitches is easiest to detect while working with the features used in this algorithm and hardly confused with any other emotion. However, the algorithm seems to classify Disgust emotion as Angry more than half of the times. It needs to be pointed out that even humans may not be able to distinguish disgust and anger solely through voice without the assistance of visual intakes.

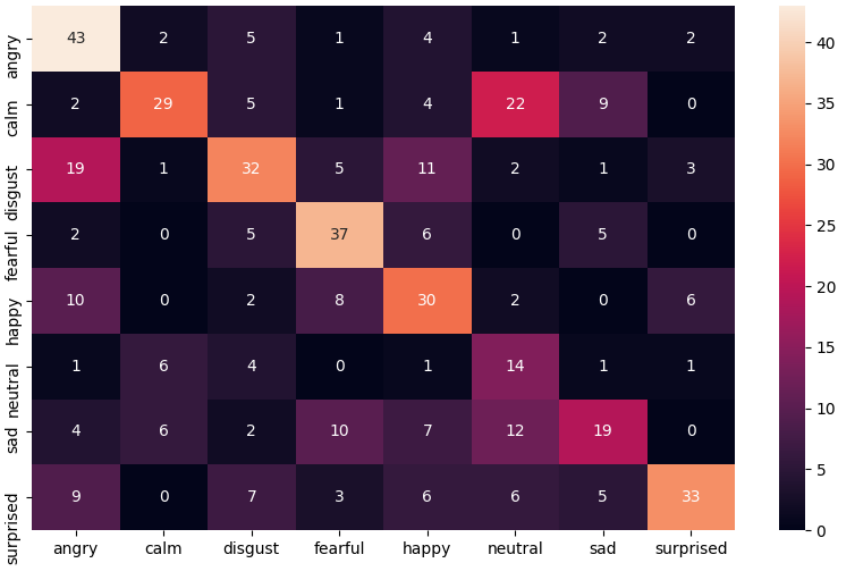

Figure 4.12. Confusion Matrix for Deep Neural Network

For the next step, MFCC feature is removed which unexpectedly, does not affect the accuracy considerably. Total accuracy of the model only dropped by less than 2% and settled on 89.94% for seven primary emotions. In another test, it only drops to 90.16%, hardly affected by the change. Its confusion matrix in Figure 4.13 reveals more

interesting details. By removing MFCC, Calm state turns out to be most correctly predicted label, followed by Fear emotion which pushes back the category of Anger to the third place. However, the removal makes the "happy class" almost unpredictable as it is categorized mostly as surprised and angry. All the emotions, other than Calm, appear to suffer from the absence of MFCC feature, while the improvement in prediction calm prevent the accuracy from decreasing severely.

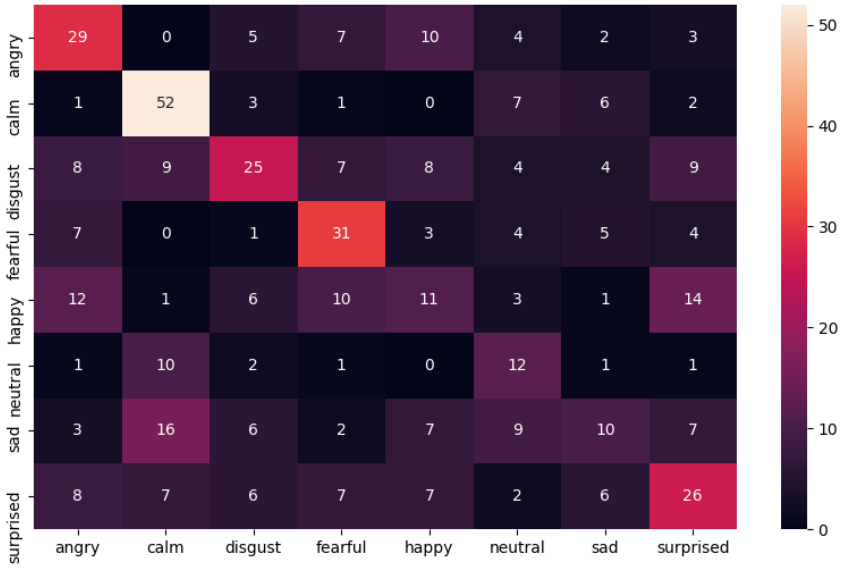

Figure 4.13. DNN confusion matrix without MFCC

The next feature to be removed, is Mel Spectrogram. The accuracy reacts in an unexpected way to this action. Instead of decreasing, it increases. While the accuracy for all the three features combined is 90.57%, after eliminating Mel Spectrogram feature from computations, it increases to 92.03%. The change is not remarkable, but it provides a solid fact: The DNN algorithm works well when it is free from the limits caused by Mel Spectrogram feature. It can evenly provide accuracy for all the seven emotions, not just majority of them.

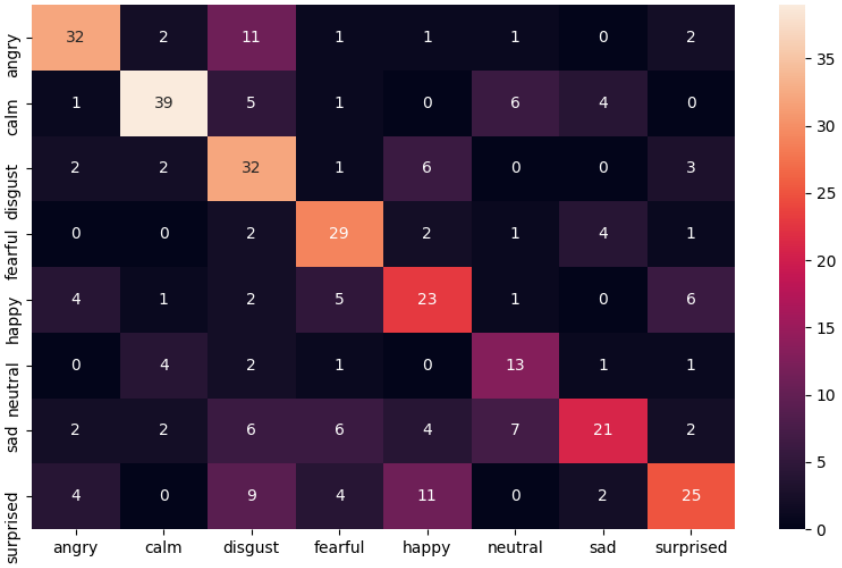

Figure 4.14. DNN confusion matrix without Mel Spectrogram

In the final run for comparison, Chroma feature is eliminated from the algorithm to study its importance and effects on the performance of the model. As appeared in Figure 4.15 without any need for further details, there are obvious improvements in classification of emotions. The accuracy once again does not decrease but increases to 95.12%. There are still drawbacks considering Surprised and Angry state of emotion, however, the progress in detecting the rest covers this shortcoming. Similar to the results from removing the other two features, Calm category experiences the most improvement. I believe the reason behind this increase in accuracy, which is more than all the other accuracies, features removed or the three of them combined, is that the harmony between features matters and it seems that MFCC and Mel Spectrogram work more efficiently together and cover the flaws of the other.

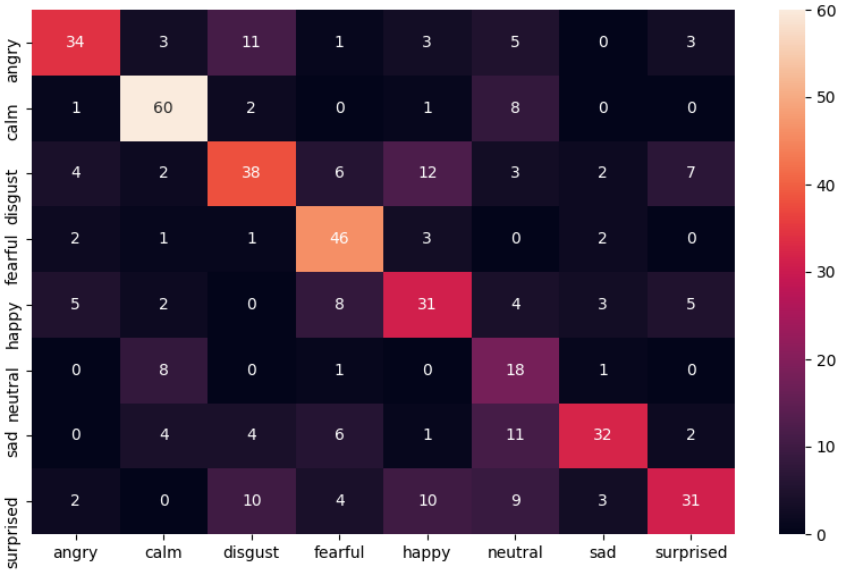

Figure 4.15. DNN confusion matrix without Chroma

4.4. Conclusion

In conclusion, the results show that the features chosen in this study do not improve the performance of Deep Sequential Neural Network. MFCC, Mel and Chroma spectrogram are not the suitable features to be combined and applied in this algorithm, since with any of them removed, the accuracy is neither affected nor improved. For better comparison, the results obtained from various combinations are presented in Figure 3.24 below.

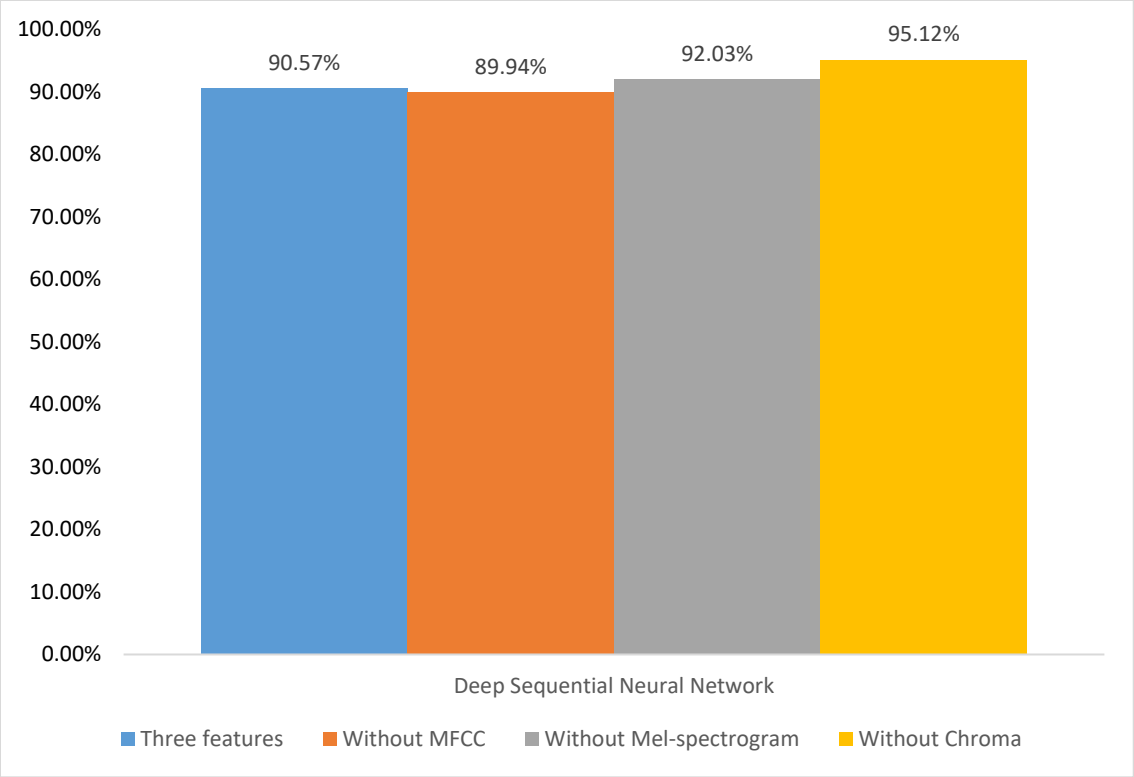

Figure 4.16 Accuracies for Deep Sequential Neural Network

Additionally, the reason I removed features one by one instead of having the model run with one of them at time, is to find the best combination and study the efficiency of features, once are together. It is obvious that while MFCC and Mel Spectrogram provides higher accuracy, Mel Spectrogram and Chroma are the worst combination and the three features together is not a combination to be considered.

Chapter 5.

Conclusion and Future Directions

5.1. Introduction

The initial purpose of this study is to observe the performances of two algorithms, one classical machine learning and one deep network in the field of Emotion Recognition through voice. Throughout the study two algorithms are developed, trying to hold the minimum differences regarding other circumstances for the comparison to be as accurate as possible. Therefore, not only is the dataset the same for both of algorithms, but features are used in the Deep Learning algorithm to keep it more limited. The dataset is The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS), the visual and song files disregarded for this study. It contains both female and male voices performing seven primary emotions, Anger, Calm, Disgust, Fear, Happy, Sad, Surprised in addition to a Neutral state. The features are MFCC, Mel and Chroma spectrogram.

Throughout the study, another step is considered to deliver a more comprehensive conclusion. The features are removed one by one to determine the best combination for each algorithm. In this chapter, those results are being compared and analyzed as well.

In order to both study the performance of algorithms on each emotion and observe the impact of removing features also on the emotions, a confusion matrix is attached to the codes. Four confusion matrices are studied and compared in this chapter.

By the end of the chapter, conclusion and a path for future adjustments and improvements are presented.

5.2. Comparison

These codes are run on a Surface Laptop 3 with an Intel(R) Core i7-1065G7 CPU. The operating system is Windows 10, version 20H2. The codes are in python

language, version 3.9, and performed on PyCharm. The information is mentioned as they are deciding factors that determine run-time, which also is important for comparison and conclusion section.

The comparison starts with algorithms lacking one specific feature and, in the end, comes to comparing two complete methods, with all three features and their run-times.

5.2.1. Algorithms without MFCC

Removing MFCC, the accuracies for both algorithms decrease. However, while the accuracy of MLP classifier decreases more dramatically to 37.50%, the accuracy of Deep Sequential Neural Network is slightly affected and dropped to 89.94%, proving the point that a Deep Neural Network works perfectly fine without the feature extraction step. The confusion matrices for both algorithms are presented in Figure 5.1 side by side.

Even though there is a decrease in both accuracies, the performance of DNN is still considerably better than the one of MLP classifier. They both detect Calm state of emotion more correctly than the rest, while the number for DNN is 52 and 33 for MLP classifier, let alone almost all the other emotions which are recognised at least slightly better by DNN. Happy and Sad categories are exceptions. Even without the help of MFCC, MLP classifier detects them correctly more often than DNN algorithm. However, it does not help outperforming the Deep algorithm, as it does not cover the difference between performances regarding other categories. Mostly, Disgust class is almost unrecognizable without MFCC in MLP classifier, while perfectly detectable by DNN, without the help of MFCC feature.

To summarize the comparison, I believe the color bar next to each confusion matrix is enough for commenting on the performances. The color white represents the highest number detection, correct or incorrect though. For MLP classifier, the color is associated with number 30 or higher, while for DNN, the number is 50 and higher. The white blocks in both algorithms are in diagonal line which are blocks for correctly recognised. However, the one for DNN shows number 52 and the one for MLP classifier shows number 33. Other numbers in diagonal, as mentioned and observed above, prove the point as well.

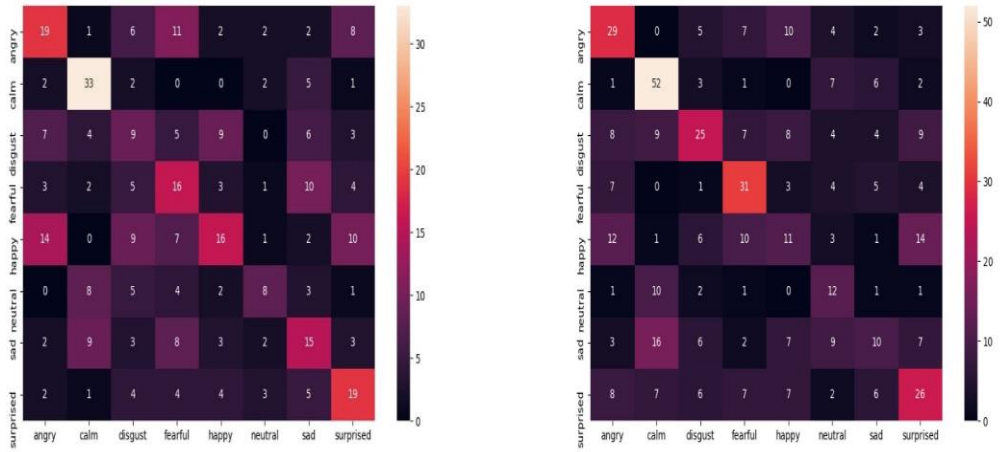

Figure 5.1. MLP Classifier (left) and DNN (right) without MFCC

5.2.2. Algorithms without Mel spectrogram

The removal of Mel spectrogram feature does affect both algorithms almost the same amount, however, not in the same direction. While the accuracy of MLP classifier decreases nearly 2% from 42.38% to 40.83%, the accuracy of Deep Sequential Neural Network increases next to 2% from 90.57% to 92.03%.

The performance of DNN significantly improves as shown in the confusion matrix, in which the majority of all seven categories are detected correctly. Its confusion matrix shows consistent recognition among seven classes, whereas the confusion matrix of MLP classifier implies random categorization instead of methodological approach towards the task. Neutral class is unrecognizable, and fearful and sad classes are barely distinguished from each other.

The color bars alongside the matrices does not significantly outshine the other. The rate for DNN is slightly higher with 35 and above, while the other bar shows there are only 30 and higher in each category at the best. The highest numbers detected in both matrices belongs to the diagonal line, which means they are categorized correctly and neither one of the algorithms outperform the other in this regard.

However, what provides the higher accuracy for DNN algorithm, seems to be the consistency. The best detection number in both algorithms, angry in MLP classifier with

number 32 and Calm in DNN with number 39, are close, while the least correct recognitions are considerably distinct, let alone the other classes in between. The difference in numbers is not significant if categories are compared one by one. Although DNN outperform its non-deep competitor in all of them, the gap could be overlooked. However, when the comparison is overall and gaps are combined, they result in significant difference in accuracy.

In summary, as in Figure 5.2, the confusion matrix of DNN without Mel Spectrogram shows an obvious order which lead to the higher accuracy. While the other confusion does not follow any specific pattern, hence, the low accuracy is predictable.

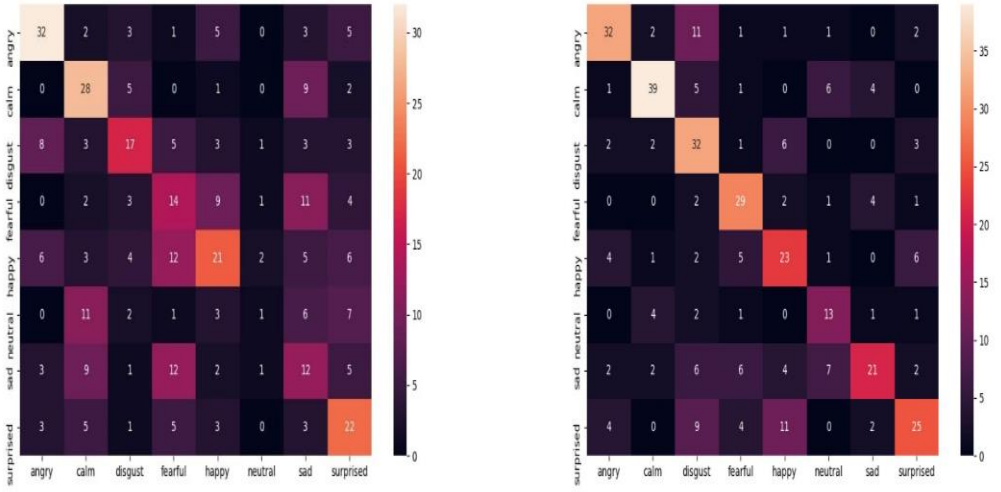

Figure 5.2. MLP Classifier (left) and DNN (right) without Mel Spectrogram

5.2.3. Algorithms without Chroma

The two algorithms react remarkably different to the removal of Chroma feature. There is not only the fact that MLP classifier drops almost two percent of accuracy and decreases to 40.83%, but also the significant improvement in DNN, accuracy-wise. The accuracy of Deep Neural Network increases from 90.57% to 95.12%, showing next to 5% of improvement in performance. Both of algorithms detects Calm state of emotion the most accurately, however, there is a difference that is not to be ignored. While the number for MLP classifier is 26 (in the confusion matrix, alongside its Angry category), the number for DNN is 60, higher than any other number so far. Going through the row

and noticing the highest confused number of data is 8 that are mistaken for Neutral category, indicates that the confusion rates are not even closed. Moreover, the lowest detection rate in DNN which is Neutral class by 18, stands among the average number of detections in MLP classifier.

It is obvious from the comparison between color bar alongside the matrices. While the highest number MLP classifier could categorize in the same class appears to be 25 and higher, the bar for DNN goes all the way up to the 60. The deep purple blocks which are associated with low numbers are all outside diagonal line of DNN confusion matrix, while scattered all over the confusion matrix of MLP classifier. Without knowing the exact percentage of accuracy, it is predictable with a glance through the matrices that one is fallowing an obvious pattern whereas the other one is disordered.

In conclusion, Chroma feature proves to be of help for MLP classifier which is not supported by deep learning and numbers of hidden layer. However, removing the feature helps Deep Sequential Neural Network to learn more effectively and improves the accuracy. Specifically, and most obviously, DNN does not require Chroma feature to detect Calm state of emotion while the feature seems to work perfectly fine for MLP classifier.

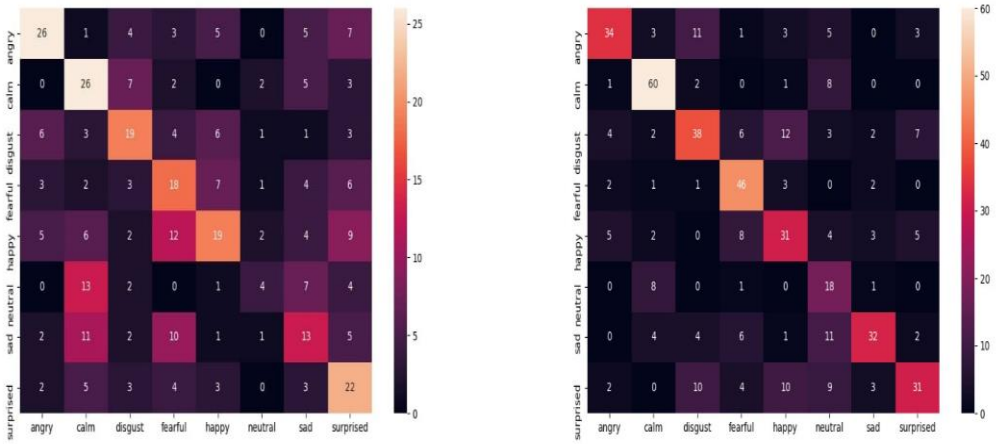

Figure 5.3. MLP Classifier (left) and DNN (right) without Chroma

5.3. Conclusion

Figure 5.4 below presents all the accuracies achieved throughout this study. For better comparison, they are gathered alongside each other. First row includes simulations of Deep Sequential Neural Network and the bars in the next row are simulations on MLP classifier. Blue bars are the initial results, the orange bars show accuracies for each algorithm when MFCC is removed, the gray bars represent accuracies without Mel spectrogram, and at last, the yellow bars belong to accuracies of models without Chroma feature.

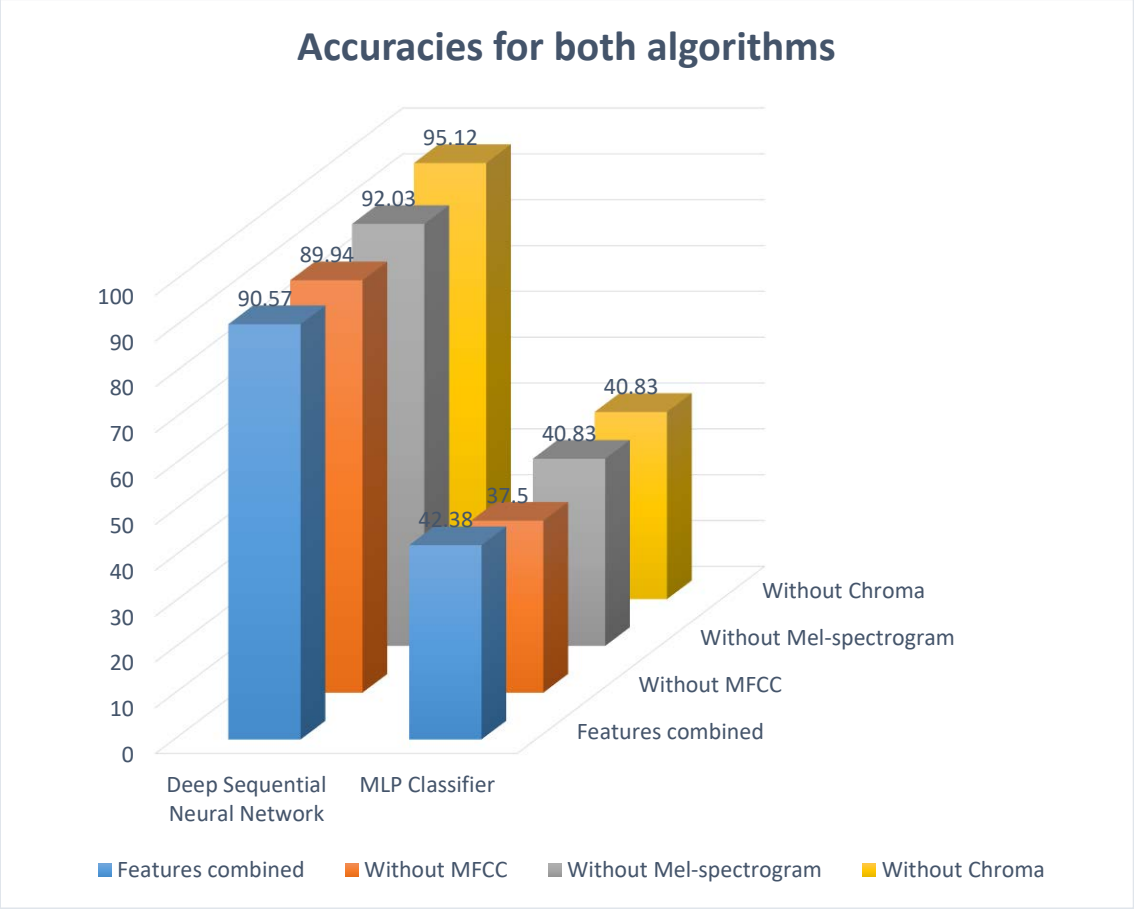

Figure 5.4. Final results for all simulations

As from results summarized by Figure 5.4, it becomes clear that Deep Sequential Neural Network, which is not a highly complicated Deep Neural Network, performs better without the restriction of unnecessary features. It performs the best without Chroma feature that is not initially a popular feature for Speech Emotion Recognition. The study suggests even though Chroma improves the accuracy of a MLP classifier, it does not sit

well alongside Mel Spectrogram and MFCC for Deep Neural Networks. Using Chroma makes distinguishing Calm, Neutral and Sad category from one another a more challenging task. Moreover, even while chroma increases the accuracy of MLP classifier, the method is still struggling to detect aforementioned classes. It seems that the best approach toward detection of Calm, Neutral and Sad voices is to let the Deep Neural Network train itself.

Regarding the comparison between the two methods, other than the advantage in run-time, DNN outperforms MLP classifier in every possible aspect. It is to point out that only training the network is taking longer than the MLP classifier. Once the model is generated, a real-time experiment might deliver these more accurate results in a lesser time and the only advantage of MLP classifier would vanish.

However, since MLP classifiers are mostly utilized to fuse other stronger models together within their layers, this study serves to provide potential features that sit well with the main frame of an algorithm. It also studies the efficiency of Chroma in SER and proved its usability for a MLP classifier.

5.4. Future direction

As implementing more features appears to be helpful with regards to accuracy of MLP classifier, and the algorithm mostly faces difficulties in recognizing Neutral and Calm states, for future works a feature which is not focused on pitches or melodies could deliver better results. Another suggestion is to combine a Speech Emotion Recognition algorithm with high accuracy, in this case, DNN, with an algorithm that recognizes emotions through image and delivers acceptable results, solely for the purpose of distinguishing Neutral and Calm state from Sad.

On the other hand, DNN seems to work most efficiently with Mel Spectrogram and MFCC. However, since the removal of Mel Spectrogram increases the accuracy, MFCC might be enough as well. This act might also decrease the run-time for DNN algorithm.

Another future path could be implementing the algorithms on a social robot (e.g., NAO robots) and carrying out real-time experiments. As the model is ready, the flowchart of implementation would be as presented in Figure 5.5.

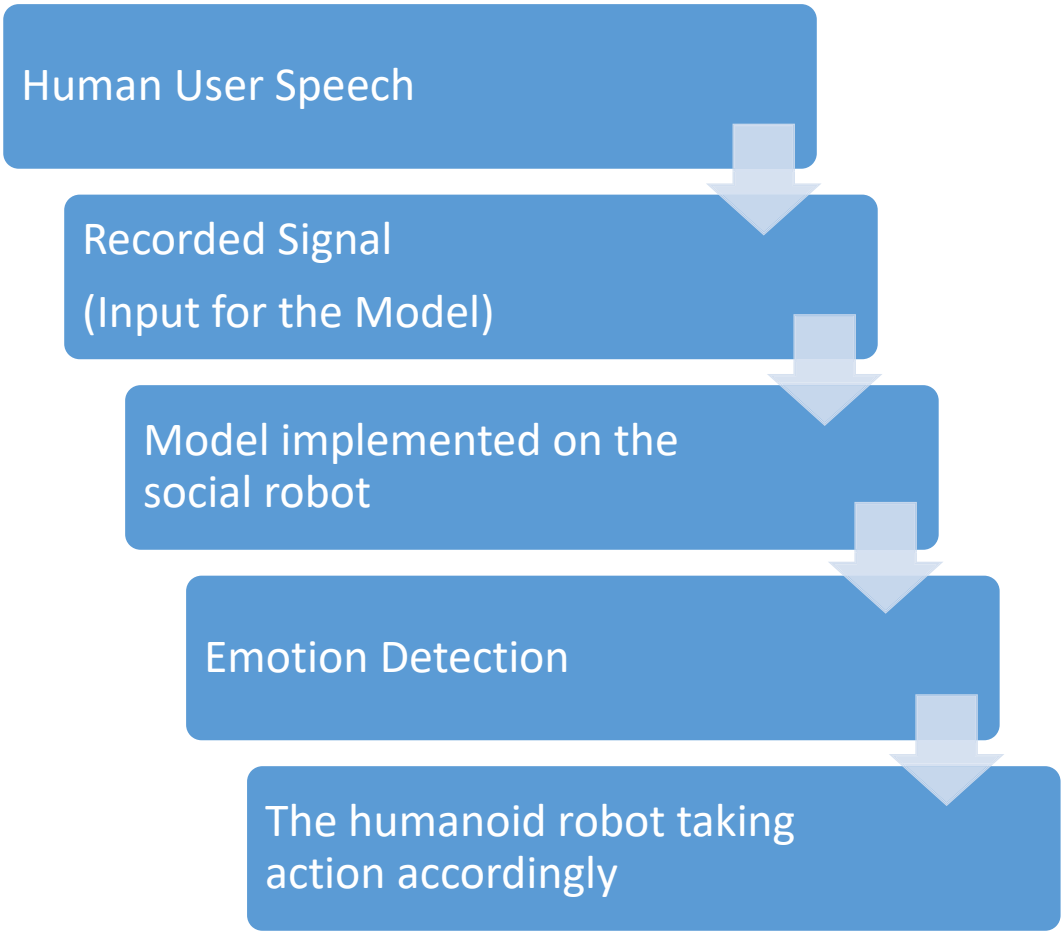

Figure 5.5. Implementing a model for Emotion Recognition by voice on a social robot

As the model is supposedly already trained, the one drawback of Deep Sequential Neural Network regarding its training time is not an issue anymore. That is what makes the aforementioned model a better suit for virtual assistants and social robots. Another reason is that even though this algorithm is a Deep Neural Network with high accuracy, it is not computationally complex. Hence, I presume the processors of any social robot would not face difficulties processing the algorithm. That is why it would be another suitable approach towards future advancements in this field.

References

[1] B. Gates, "A Robot in Every Home," *Scientific American,* vol. 296, no. 1, pp. 58-68, February 2007.

[2] R. W. Picard, "Affective Computing," MIT Press, 1997.

[3] R. W. Picard, "Affective Computing-MIT Media Laboratory Perceptual Computing Section Technical Report No. 321," *Cambridge, MA,* vol. 2139, 1995.

[4] J. a. T. T. Tao, "Affective computing: A review," in *Affective Computing and Intelligent Interaction*, vol. 3784, Berlin, Springer, 2005, pp. 981-995.

[5] S. a. C. E. a. B. R. a. H. A. Poria, "A review of affective computing: From unimodal analysis to multimodal fusion," *Information Fusion,* vol. 37, pp. 98--125, 2017.

[6] K. R. Scherer, "Psychological Models of Emotion," in *The Neuropsychology of Emotion*, USA, Oxford University Press, 2000, May 18, 2000, p. 536.

[7] R. C. a. M. M. M. a. C. G. L. Sinclair, "Mood-related persuasion depends on (mis) attributions," *Guilford Press,* vol. 12, no. 4, pp. 309-326, 1994.

[8] T. T. Ortony A, "What's basic about basic emotions?," *Psychological review,* vol. 97, no. 3, p. 315, 1990.

[9] A. LIM, "Design and Implementation of Emotions for Humanoid Robots based on the Modality-independent DESIRE Model," Department of Intelligence Science Graduate School of Informatics Kyoto University, Kyoto - Japan, 2012.

[10] E. Cambria, A. Livingstone and A. Hussain, "The Hourglass of Emotions," Berlin, Springer, 2012.

[11] A. D. a. S. C. C. Dileep, "GMM-based intermediate matching kernel for classification of varying length patterns of long duration speech using support

vector machines," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 25, no. 8, pp. 1421-1432, 2013.

[12] D. a. E. K. a. L. K. Neiberg, "Emotion recognition in spontaneous speech using GMMs," in *Ninth international conference on spoken language processing*, 2006.

[13] A. D. a. S. C. C. Dileep, "HMM based intermediate matching kernel for classification of sequential patterns of speech using support vector machines," *IEEE transactions on audio, speech, and language processing,* vol. 21, no. 12, pp. 2570-2582, 2013.

[14] G. a. D. M. K. a. R. K. a. P. J. a. o. Vyas, "An automatic emotion recognizer using MFCCs and Hidden Markov Models," in *2015 7th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, Brno, Czech Republic , 2015.

[15] Y. Bengio, Learning deep architectures for AI, Now Publishers Inc, 2009.

[16] R. A. a. J. E. a. B. M. I. a. J. T. a. Z. M. H. a. A. T. Khalil, "Speech emotion recognition using deep learning techniques: A review," *IEEE,* vol. 7, pp. 117327-117345, 2019.

[17] V. a. P. P. Chernykh, "Emotion recognition from speech with recurrent neural networks," *arXiv preprint arXiv:1701.08071,* 2017.

[18] J. R. Bellegarda, "Large-scale personal assistant technology deployment: the siri experience.," in *INTERSPEECH*, 2013.

[19] I. a. R. K. a. K. I. a. R. K. a. C. K. a. W. H. a. S. P. a. H. D. a. L. Q. a. M. A. Lopatovska, "Talk to me: Exploring user interactions with the Amazon Alexa," *Journal of Librarianship and Information Science,* vol. 51, no. 4, pp. 984-997, 2019.

[20] "SoftBank robotics," SoftBank robotics , [Online]. Available: https://www.softbankrobotics.com/emea/en/nao.

[21] H. a. W. C. a. P. V. Binali, "Computational approaches for emotion detection in text," in *IEEE*, 2010.

[22] K. D. M. R. J. e. a. Sailunaz, "Emotion detection from text and speech: a survey," *Social Network Analysis and Mining,* vol. 8, no. 1, 2018.

[23] R. A. a. J. E. a. B. M. I. a. J. T. a. Z. M. H. a. A. T. Khalil, "Speech Emotion Recognition using Deep Learning techniques: A review," *IEEE,* vol. 7, pp. 117327-117345, 2019.

[24] S. K. a. K. J. D'mello, "A review and meta-analysis of multimodal affect detection systems," *ACM computing surveys (CSUR),* vol. 47, no. 3, pp. 1-36, 2015.

[25] G. Kataria, A. Gupta, V. S. Kaushik and G. Chaudhary, "Emotion Recognition from Speech Signals Using Machine Learning and Deep Learning Techniques," in *Concepts and Real-Time Applications of Deep Learning*, Springer International Publishing, 2021, pp. 63-73.

[26] B. C. Ko, "A brief review of facial emotion recognition based on visual information," *sensors,* vol. 18, no. 2, 2018.

[27] J. B. Watson, "Psychological care of infant and child," *W W Norton & Co,* 1928.

[28] M. B. Arnold, " Emotion and personality," *Psychological aspects,* vol. 1, 1960.

[29] J. L. a. R. D. Tracy, "Four Models of Basic Emotions: A Review of Ekman and Cordaro, Izard, Levenson, and Panksepp and Watt," *Emotion Review,* vol. 3, no. 4, pp. 397-405, 2011.

[30] S. S. Tomkins, "Affect theory," *Approaches to emotion,* vol. 163, no. 163-195, 1984.

[31] I. J. Roseman, "Cognitive determinants of emotion: A structural theory.," *Review of Personality & Social Psychology,* vol. 5, pp. 11-36, 1984.

[32] S. Seconds, "Plutchik's Wheel of Emotions: Exploring the Emotion Wheel," 11 August 2020. [Online]. Available: https://www.6seconds.org/2020/08/11/plutchik-wheel-emotions/.

[33] J. A. Russell, "A circumplex model of affect," *Journal of personality and social psychology,* vol. 39, no. 6, p. 1161, 1980.

[34] P. Ekman, "Expression and the nature of emotion," *Approaches to emotion,* vol. 3, no. 19, p. 344, 1984.

[35] K. R. a. o. Scherer, "On the nature and function of emotion: A component process approach," *Approaches to emotion,* vol. 2293, no. 317, p. 31, 1984.

[36] B. a. G. S. Weiner, "An attributional approach to emotional development," *Emotions, Cognition, and Behavior,* pp. 167-191, 1984.

[37] N. H. a. o. Frijda, The Emotions, Cambridge University Press, 1986.

[38] P. a. S. J. a. K. D. a. O. C. Shaver, "Emotion knowledge: further exploration of a prototype approach," *Journal of personality and social psychology,* vol. 52, no. 6, p. 1061, 1987.

[39] K. a. J.-L. P. N. Oatley, "Towards a cognitive theory of emotions," *Cognition and emotion,* vol. 1, no. 1, pp. 29-50, 1987.

[40] G. L. a. O. A. Clore, "Psychological construction in the OCC model of emotion," *Emotion Review,* vol. 5, no. 4, pp. 335-343, 2013.

[41] C. a. L. R. Smith, "Emotion and Adaptation," *Handbook of personality: Theory and research,* pp. 609-637, 1990.

[42] R. S. Lazarus, "Emotion theory and psychotherapy," *Emotion, psychotherapy, and change,* pp. 290-301, 1991.

[43] H. Lovheim, "A new three-dimensional model for emotions and monoamine neurotransmitters," *Medical hypotheses,* vol. 78, no. 2, pp. 341-348, 2012.

[44] L. A. H. A. Cambria E., "The Hourglass of Emotions," in *Cognitive behavioural systems*, vol. 7403, Springer, 2012, pp. 144-157.

[45] K. R. Scherer, "Psychological Models of Emotion," in *The Neuropsychology of Emotion*, New York, OXFORD university press, 2000, pp. 137-162.

[46] A. Mehrabian, Silent Messages, Belmont: Wadsworth Publishing Company, 1971.

[47] M. J. Bennett, Basic Concepts of Intercultural Communication: Selected Readings., ERIC, 1998.

[48] N. a. C. I. a. H. T. S. Sebe, "Multimodal Emotion Recognition," in *Handbook of pattern recognition and computer vision*, World Scientific, 2005, pp. 387-409.

[49] K. a. o. Takahashi, "Remarks on emotion recognition from bio-potential signals," in *Citeseer*, 2004.

[50] M. a. W. F. a. K. T. a. S. B. a. S. C. a. S. K. a. M. L.-P. W{\"o}llmer, "Youtube movie reviews: Sentiment analysis in an audio-visual context," *IEEE Intelligent Systems,* vol. 28, no. 3, pp. 46-53, 2013.

[51] P. a. K. D. Ekman, "Universal facial expressions of emotion," *Segerstrale U, P. Molnar P, eds. Nonverbal communication: Where nature meets culture,* vol. 27, p. 46, 1997.

[52] P. a. F. W. Ekman, Facial Action Coding System, Consulting Psychologists Press, 1978.

[53] A. A. a. D. N. V. Dixit, "A Survey on detection of reasons behind infant cry using speech processing," in *International Conference on Communication and Signal Processing (ICCSP)*, 2018.

[54] P. a. S. J. a. K. D. a. O. C. Shaver, "Emotion knowledge: further exploration of a prototype approach," *Journal of personality and social psychology,* vol. 52, no. 6, p. 1061, 1987.

[55] N. Campbell, "Databases of emotional speech," in *ISCA Tutorial and Research Workshop (ITRW) on Speech and Emotion*, 2000.

[56] E. a. C. R. a. S. I. a. C. C. a. L. O. a. M. M. a. M. J.-C. a. D. L. a. A. S. a. B. A. a. o. Douglas-Cowie, "The HUMAINE database: Addressing the collection and annotation of naturalistic and induced emotional data," in *International conference on affective computing and intelligent interaction*, Springer, 2007, pp. 488-500.

[57] O. a. K. I. a. M. B. a. P. I. Martin, "The eNTERFACE'05 audio-visual emotion database," in *22nd International Conference on Data Engineering Workshops (ICDEW'06)*, IEEE, 2006, pp. 8-8.

[58] M. {. a. F. {. a. T. {. a. B. {. a. C. {. a. K. {. a. L. {Morency}, "YouTube Movie Reviews: Sentiment Analysis in an Audio-Visual Context," *IEEE Intelligent Systems,* vol. 28, no. 3, pp. 46-53, May 2013.

[59] L.-P. a. M. R. a. D. P. Morency, "Towards Multimodal Sentiment Analysis: Harvesting Opinions from the Web," in *Proceedings of the 13th International Conference on Multimodal Interfaces*, New York, NY, USA, ACM, 2011, pp. 169-176.

[60] F. a. P. A. a. R. M. a. S. W. F. a. W. B. Burkhardt, "A database of German emotional speech," in *Ninth European Conference on Speech Communication and Technology*, 2005.

[61] I. S. a. H. A. V. a. A. O. a. D. P. Engberg, "Design, recording and verification of a Danish emotional speech database," in *Fifth European Conference on Speech Communication and Technology*, 1997.

[62] V. a. K. Z. a. M. A. a. B. A. a. N. A. Hozjan, "Interface Databases: Design and Collection of a Multilingual Emotional Speech Database," in *LREC*, 2002.

[63] A. a. H. C. a. S. S. a. N. E. a. D. S. a. R. M. J. a. W. M. Batliner, "" You Stupid Tin Box"-Children Interacting with the AIBO Robot: A Cross-linguistic Emotional Speech Corpus," in *Lrec*, 2004.

[64] B. a. R. G. a. L. M. Schuller, "Speech emotion recognition combining acoustic features and linguistic information in a hybrid support vector machine-belief network architecture," *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing,* vol. 1, pp. I--577, 2004.

[65] J. a. B.-G. S. Hansen, "Getting started with SUSAS: A speech under simulated and actual stress database," in *Fifth European Conference on Speech Communication and Technology*, Rhodes, 1997.

[66] S. a. S. S. a. L. X. a. B. J. Yacoub, "Recognition of emotions in interactive voice response systems," in *Eighth European conference on speech communication and technology*, 2003.

[67] M. a. M. G. Slaney, "BabyEars: A recognition system for affective vocalizations," *Speech Communication,* vol. 39, no. 3-4, pp. 367-384, 2003.

[68] D. a. E. K. Neiberg, "Automatic recognition of anger in spontaneous speech," in *Ninth Annual Conference of the International Speech Communication Association*, 2008.

[69] C. a. B. M. a. L. C.-C. a. K. A. a. M. E. a. K. S. a. C. J. N. a. L. S. a. N. S. S. Busso, "IEMOCAP: interactive emotional dyadic motion capture database," *Language Resources and Evaluation,* vol. 42, no. 4, p. 335, 05 November 2008.

[70] M. K. P.-F. Kate Dupuis, "University of Toronto," University of Toronto, Psychology Department, 21 June 2010. [Online]. Available: https://tspace.library.utoronto.ca/handle/1807/24487.

[71] J. M. a. G.-A. J. a. C. J. a. E. E. a. P. J. M. Montero, "Analysis and modelling of emotional speech in Spanish," in *Proc. of ICPhS*, 1999.

[72] N. a. R. S. a. L. N. Amir, "Analysis of an emotional speech corpus in Hebrew based on objective criteria," in *ISCA Tutorial and Research Workshop (ITRW) on Speech and Emotion*, 2000.

[73] M. Schroder, "Experimental study of affect bursts," *Speech communication,* vol. 40, no. 1-2, pp. 99-116, 2003.

[74] V. a. P. V. A. Makarova, "RUSLANA: A database of Russian emotional utterances," in *Seventh international conference on spoken language processing*, 2002.

[75] E. a. C. G. a. B. A. a. S. M. a. S. B. W. Parada-Cabaleiro, "DEMoS: an Italian emotional speech corpus," *Language Resources and Evaluation,* vol. 54, no. 2, pp. 341-383, 2020.

[76] K. R. a. G. D. a. J. T. a. K. G. a. B. T. Scherer, "Acoustic correlates of task load and stress," in *Seventh international conference on spoken language processing*, 2002.

[77] R. a. S. R. a. K. R. a. P. J. M. Tato, "Emotional space improves emotion recognition," in *Seventh International Conference on Spoken Language Processing*, 2002.

[78] E. M. a. C. P. a. D. C. a. T. G. a. C. F. Caldognetto, "Modifications of phonetic labial targets in emotive speech: effects of the co-production of speech and emotions," *Speech Communication,* vol. 44, no. 1-4, pp. 173-185, 2004.

[79] C. M. a. N. S. S. Lee, "Toward detecting emotions in spoken dialogs," *IEEE transactions on speech and audio processing,* vol. 13, no. 2, pp. 293-303, 2005.

[80] S. R. A. R. F. A. Livingstone, "The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English," *PLOS ONE,* vol. 13, no. 5, pp. 1-35, May 2018.

[81] W. a. Z. T. F. a. X. M.-X. a. B. H.-J. Wu, "Study on speaker verification on emotional speech," in *Ninth International Conference on Spoken Language Processing*, 2006.

[82] M. a. M. E. a. K. K. a. N. S. Grimm, "Combining categorical and primitives-based emotion recognition," in *14th European Signal Processing Conference*, 2006.

[83] S. T. a. K. Z. a. D. M. a. R. M. Jovicic, "Serbian emotional speech database: design, processing and evaluation," in *9th Conference Speech and Computer*, 2004.

[84] R. a. P. R. W. Fernandez, "Modeling drivers' speech under stress," *Speech communication,* vol. 40, no. 1-2, pp. 145-159, 2003.

[85] M. a. K. K. a. N. S. Grimm, "The Vera am Mittag German audio-visual emotional speech database," in *IEEE international conference on multimedia and expo*, 2008.

[86] A. a. E. Z. Harimi, "A database for automatic Persian speech emotion recognition: collection, processing and evaluation," *International Journal of Engineering,* vol. 27, no. 1, pp. 79-90, 2014.

[87] S. a. O. Z. a. Z. R. a. H. L. Klaylat, "Arabic Natural Audio Dataset," *Mendeley Data,* 2018.

[88] A. a. M. E. a. C. G. a. T. M. a. B. B. a. B. M. a. D. N. C. Mencattini, "Speech emotion recognition using amplitude modulation parameters and a combined feature selection procedure," *Knowledge-Based Systems,* vol. 63, pp. 68-81, 2014.

[89] A. a. U. S. a. M. W. Schmitt, "A Parameterized and Annotated Spoken Dialog Corpus of the CMU Let's Go Bus Information System," in *LREC*, 2012.

[90] H. a. S. T. a. N. M. a. K. H. Mori, "Constructing a spoken dialogue corpus for studying paralinguistic information in expressive conversation and analyzing its

statistical/acoustic characteristics," *Speech Communication,* vol. 53, no. 1, pp. 36-50, 2011.

[91] M. K. Pichora-Fuller and K. Dupuis, "Toronto emotional speech set (TESS)," Scholars Portal Dataverse, 2020.

[92] M. a. K. M. S. a. K. F. El Ayadi, "Survey on speech emotion recognition: Features, classification schemes, and databases," *Pattern Recognition,* vol. 44, no. 3, pp. 572-587, 2011.

[93] C. E. a. S. K. N. Williams, "Vocal correlates of emotional states," *Speech evaluation in psychiatry,* pp. 221 - 240, 1981.

[94] R. a. D.-C. E. a. T. N. a. V. G. a. K. S. a. F. W. a. T. J. Cowie, "Emotion recognition in human-computer interaction," *Signal Processing Magazine, IEEE,* vol. 18, pp. 32-80, 2001.

[95] A. J. Murray IR, "Toward the simulation of emotion in synthetic speech: a review of the literature on human vocal emotion," *The Journal of the Acoustical Society of America,* vol. 93, no. 2, p. 1097–1108, 1993.

[96] T. L. a. F. S. W. a. D. S. L. C. Nwe, "Speech emotion recognition using hidden Markov models," *Speech communication,* vol. 41, no. 4, pp. 603-623, 2003.

[97] L. B. R., "Enhancement of noisy speech signals: Application to mobile radio communications," *Speech Communication,* vol. 18, no. 1, pp. 3-19, 1996.

[98] S. E. a. H. J. H. Bou-Ghazale, "A comparative study of traditional and newly proposed features for recognition of speech under stress," *IEEE Transactions on speech and audio processing,* vol. 8, no. 4, pp. 429-442, 2000.

[99] H. a. T. S. Teager, "Evidence for nonlinear sound production mechanisms in the vocal tract," in *Speech production and speech modelling*, Springer, 1990, pp. 241-261.

[100] T. H., "Some observations on oral air flow during phonation," *IEEE Transactions on Acoustics, Speech, and Signal Processing,* vol. 28, no. 5, pp. 599-601, 1980.

[101] J. F. Kaiser, "On a simple algorithm to calculate the'energy'of a signal," in *International conference on acoustics, speech, and signal processing*, IEEE, 1990, pp. 381-384.

[102] J. H. a. B.-G. S. E. a. S. R. a. P. B. Hansen, "Getting started with SUSAS: a speech under simulated and actual stress database," in *Eurospeech*, 1997.

[103] J. T. K. a. N. R. Nicholson, "Emotion recognition in speech using neural networks," *Neural computing & applications (Springer),* vol. 9, no. 4, pp. 290--296, 2000.

[104] S. a. D. S. Ramamohan, "Sinusoidal model-based analysis and classification of stressed speech," *IEEE Transactions on audio, speech, and language processing,* vol. 14, no. 3, pp. 737--746, 2006.

[105] S. G. a. M. S. a. K. V. A. a. C. S. a. R. K. S. Koolagudi, "IITKGP-SESC: speech database for emotion analysis," in *International conference on contemporary computing*, Berlin, Heidelberg, 2009.

[106] V. Petrushin, "Emotion in speech: Recognition and application to call centers," in *Proceedings of artificial neural networks in engineering*, 1999.

[107] T. a. A. C.-N. Iliou, "Statistical evaluation of speech features for emotion recognition," in *2009 Fourth International Conference on Digital Telecommunications*, IEEE, 2009, pp. 121--126.

[108] A. B. a. R. A. a. B. T. K. Kandali, "Emotion recognition from speeches of some native languages of Assam independent of text and speaker," in *National Seminar on Devices, Circuits and Communication, Department of ECE, BIT Mesra*, Mesra, 2008.

[109] F. a. V. V. K. a. J. A. a. o. Shah, "Speaker independent automatic emotion recognition from speech: a comparison of MFCCs and discrete wavelet

transforms," in *International Conference on Advances in Recent Technologies in Communication and Computing*, Kottayam, 2009.

[110] D. a. K. C. a. P. I. Ververidis, "Automatic emotional speech classification," in *IEEE international conference on acoustics, speech, and signal processing*, Montreal, 2004.

[111] T. a. A. E. Vogt, "Improving Automatic Emotion Recognition from Speech via Gender Differentiaion," in *LREC*, Genoa, 2006.

[112] O.-W. a. C. K. a. H. J. a. L. T.-W. Kwon, "Emotion recognition by speech signals," in *Eighth European conference on speech communication and technology*, Geneva, 2003.

[113] A. a. E. Z. Harimi, "A database for automatic Persian speech emotion recognition: collection, processing and evaluation," *International Journal of Engineering,* vol. 27, no. 1, pp. 79-90, 2014.

[114] J. B. a. C. J. a. M. M. a. T. C. M. Alonso, "New approach in quantification of emotional intensity from the speech signal: emotional temperature," *Expert Systems with Applications,* vol. 42, no. 27, pp. 9554-9564, 2015.

[115] Y.-L. a. W. G. Lin, "Speech emotion recognition based on HMM and SVM," in *International conference on machine learning and cybernetics*, Guangzhou, 2005.

[116] K. T. a. M. R. J. Assaleh, "New LP-derived features for speaker identification," *IEEE Transactions on Speech and Audio Processing,* vol. 2, no. 4, pp. 630-638, 1994.

[117] J. a. Y. K. a. C. J. a. W. T. a. X. A. a. C. Y. a. Y. C. Wang, "DOA estimation of excavation devices with ELM and MUSIC-based hybrid algorithm," *Cognitive Computation,* vol. 9, no. 4, pp. 564--580, 2017.

[118] S. a. M. P. Davis, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE transactions on acoustics, speech, and signal processing,* vol. 28, no. 4, pp. 357--366, 1980.

[119] S. S. a. V. J. a. N. E. B. Stevens, "A scale for the measurement of the psychological magnitude pitch," *The journal of the acoustical society of america,* vol. 8, no. 3, pp. 185-190, 1937.

[120] H. Beigi, "Speaker recognition," in *Fundamentals of Speaker Recognition*, Springer, 2011, pp. 543--559.

[121] X. a. W. D. Zhao, "Analyzing noise robustness of MFCC and GFCC features in speaker identification," in *2013 IEEE international conference on acoustics, speech and signal processing*, Vancouver, 2013.

[122] F. N. K. G. Ganchev T, "Comparative evaluation of various MFCC implementations on the speaker verification task," in *Proceedings of the SPECOM*, Greec, 2005.

[123] M. a. K. A. N. Moinuddin, "Speaker identification based on GFCC using GMM," *International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN,* vol. 1, no. 8, 2014.

[124] D. a. B. G. J. Wang, Computational auditory scene analysis: Principles, algorithms, and applications, Wiley-IEEE press, 2006.

[125] M. a. U. S. a. S. A. Sidorov, "Emotions are a personal thing: Towards speaker-adaptive emotion recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, 2014.

[126] M. B. K. T. S. e. a. Helmstaedter, "Connectomic reconstruction of the inner plexiform layer in the mouse retina," *Nature,* vol. 500, no. 7461, p. 168–174, 2013.

[127] J. a. S. R. P. a. L. A. a. D. G. E. a. S. V. Ma, "Deep Neural Nets as a Method for Quantitative Structure–Activity Relationships," *Journal of Chemical Information and Modeling,* vol. 55, no. 2, pp. 263-274, 2015.

[128] Y. a. B. Y. a. H. G. LeCun, "Deep Learning," *Nature,* vol. 521, pp. 436-444, 28 May 2015.

[129] R. A. a. J. E. a. B. M. I. a. J. T. a. Z. M. H. a. A. T. Khalil, "Speech emotion recognition using deep learning techniques: A review," *IEEE Access,* vol. 7, pp. 117327-117345, 2019.

[130] A. Wilson, "A Brief Introduction to Unsupervised Learning," 7 December 2020. [Online]. Available: https://towardsdatascience.com/a-brief-introduction-to-unsupervised-learning-20db46445283.

[131] R. S. a. B. A. G. Sutton, Reinforcement learning: An introduction, MIT Press, 2018.

[132] X. a. B. A. a. B. Y. Glorot, "Proc. 14th International Conference on Artificial Intelligence and Statistics," 2011.

[133] Y. a. B. Y. a. H. G. LeCun, "Deep Learning," *Nature,* vol. 521, no. 7553, pp. 436-444, 27 May 2015.

[134] H. M. a. L. M. a. C. L. Fayek, "Evaluating deep learning architectures for Speech Emotion Recognition," *Neural Networks,* vol. 92, pp. 60-68, 2017.

[135] D. B. G. Ian Foster, Cloud Computing for Science and Engineering, MIT Press, 2017.

[136] S. a. G. R. a. S. G. a. A. W. a. E.-W. C. Sahu, "Adversarial auto-encoders for speech based emotion recognition," *arXiv preprint arXiv:1806.02146,* 2018.

[137] S. E. a. D. Z. a. H. W. Eskimez, "Unsupervised learning approach to feature analysis for automatic speech emotion recognition," in *IEEE International*

Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, 2018.

[138] A. a. A. J. a. R. N. a. B. S. Badshah, "Speech Emotion Recognition from Spectrograms with Deep Convolutional Neural Network," in *International Conference on Platform Technology and Service (PlatCon)*, 2017.

[139] S. Saha, "A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way," 15 December 2018. [Online]. Available: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53.

[140] J. a. M. X. a. C. L. Zhao, "Learning deep features to recognise speech emotion using merged deep CNN," *IET Signal Processing,* vol. 12, no. 6, pp. 713-721, 2018.

[141] C. W. a. S. K. Y. a. J. J. a. C. W. Y. Lee, "Convolutional attention networks for multimodal emotion recognition from speech and text data," *ACL 2018,* vol. 28, pp. 28-34, 2018.

[142] V. C. a. G. S. a. P. Prihodko, "Emotion Recognition From Speech With Recurrent Neural Networks," *ArXiv,* p. abs/1701.08071, 2017.

[143] S. a. B. E. a. Z. C. Mirsamadi, "Automatic speech emotion recognition using recurrent neural networks with local attention," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, New Orleans, 2017.

[144] E. a. Z. M. A. a. W. C. a. M. S. a. W. S. Lakomkin, "On the Robustness of Speech Emotion Recognition for Human-Robot Interaction with Deep Neural Networks," in *RSJ International Conference on Intelligent Robots and Systems (IROS 2018)*, Madrid, Spain, 2018.

[145] Y. a. S. P. a. F. P. Bengio, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks,* vol. 5, no. 2, pp. 157-166, March 1994.

[146] S. a. S. J. Hochreiter, "Long Short-Term Memory," *Neural Comput.,* vol. 9, no. 8, p. 1735–1780, 15 November 1997.

[147] S. a. T. S. a. B. H. Tripathi, "Multi-modal emotion recognition on iemocap dataset using deep learning," *arXiv preprint arXiv:1804.05788,* 2018.

[148] J. a. M. X. a. C. L. Zhao, "Speech emotion recognition using deep 1D \& 2D CNN LSTM networks," *Biomedical Signal Processing and Control,* vol. 47, pp. 312-323, 2019.

[149] X. H. a. H. H. a. L. G. a. J. S. Le, "Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting," *Water,* vol. 11, p. 1387, 07 2019.

[150] H. a. L. M. a. C. L. Fayek, "Towards real-time Speech Emotion Recognition using deep neural networks," in *9th International Conference on Signal Processing and Communication Systems (ICSPCS)*, Cairns, QLD, Australia , 2015.

[151] K. Quach, "Cosmoboffins use neural networks to build dark matter maps the easy way," 20 May 2019. [Online]. Available: https://www.theregister.com/2019/05/20/neural_networks_dark_matter/.

[152] S. a. R. R. a. Y. S. a. Q. J. a. E. J. Latif, "Transfer learning for improving speech emotion classification accuracy," *arXiv preprint arXiv:1801.06353,* 2018.

[153] W. a. Z. D. a. C. Z. a. Y. L. T. a. L. X. a. G. F. a. Y. S. Zhang, "Deep learning and SVM-based emotion recognition from Chinese speech for smart affective services," *Software: Practice and Experience,* vol. 47, no. 8, pp. 1127--1138, 2017.

[154] L. a. C. L. a. Z. D. a. Z. J. a. Z. W. Zhu, "Emotion recognition from Chinese speech for smart affective services using a combination of SVM and DBN," *Sensors,* vol. 17, no. 7, p. 1694, 2017.

[155] G. Johnson, "Theories of Emotion," Internet Encyclopedia of Philosophy, [Online]. Available: https://www.iep.utm.edu/emotion/..

[156] E. a. H. I. a. H. A. a. C. E. a. B. S. Cambria, "Sentic avatar: Multimodal affective conversational agent with common sense," in *Toward Autonomous, Adaptive, and Context-Aware Multimodal Interfaces. Theoretical and Practical Issues*, Springer, 2011, pp. 81-95.

[157] R. Plutchik and H. Kellerman, Theories of emotion, New York: Academic Press, 1980.

[158] A. a. M. C. a. E. F. a. Z. T. a. M. H.-G. a. S. B. Stuhlsatz, "Deep neural networks for acoustic emotion recognition: Raising the benchmarks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Prague, 2011.

[159] P. a. W. C. a. W. S. Barros, "Emotional expression recognition with a cross-channel convolutional neural network for human-robot interaction," in *IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, Seul, 2015.

[160] S. a. G. R. a. S. G. a. A. W. a. E.-W. C. Sahu, "Adversarial Auto-encoders for Speech Based Emotion Recognition," *arXiv preprint arXiv:1806.02146,* 2018.

[161] O. Pierre-Yves, "The production and recognition of emotions in speech: features and algorithms," *International Journal of Human-Computer Studies,* vol. 59, no. 1-2, pp. 157-183, 2003.

[162] D. a. E. K. a. L. K. Neiberg, "Emotion recognition in spontaneous speech using GMMs," in *Ninth international conference on spoken language processing*, 2006.

[163] G. a. D. M. K. a. R. K. a. P. J. a. o. Vyas, "An automatic emotion recognizer using MFCCs and Hidden Markov Models," in *IEEE*, Brno, 2015.

[164] Y. C. a. M. L. D. a. P. Yesaware, "Speech Emotion Recognition Using Support Vector Machine," *International Journal of Computer Applications,* vol. 1, no. 20, pp. 6-9, 2010.

[165] D. a. E. K. a. L. K. Neiberg, "Emotion recognition in spontaneous speech using GMMs," in *Ninth international conference on spoken language processing*, 2006.

[166] S. L. a. A. M. a. B. B. a. S. Saketh, "Speech emotion recognition," in *International Conference on Advances in Electronics Computers and Communications*, 2014.

[167] L. C. a. N. P. C. De Silva, "Bimodal emotion recognition," in *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, Grenoble, 2000.

[168] V. A. Petrushin, "Emotion recognition in speech signal: experimental study, development, and application," *studies,* vol. 3, no. 4, pp. 222-225, 2000.

[169] K. a. Y. D. a. T. I. Han, "Speech emotion recognition using deep neural network and extreme learning machine," in *Interspeech 2014*, Singapore, 2014.

[170] R. A. a. J. E. a. B. M. I. a. J. T. a. Z. M. H. a. A. T. Khalil, "Speech emotion recognition using deep learning techniques: A review," *IEEE Access,* vol. 7, pp. 117327-117345, 2019.

[171] M. K. a. D. K. Pichora-Fuller, "Toronto emotional speech set (TESS)," Scholars Portal Dataverse, Toronto, 2020.

[172] "DEWESoftX," 2021. [Online]. Available: https://manual.dewesoft.com/x/setupmodule/modules/general/math/freqdomainanalysis/stft.

[173] A. a. K. M. a. N. A. a. S. D. Shah, "Chroma Feature Extraction," in *Chroma Feature Extraction using Fourier Transform*, 2019.

[174] G. H. Wakefield, "Mathematical representation of joint time-chroma distributions," in *SPIE's International Symposium on Optical Science, Engineering, and Instrumentation*, Denver, 1999.

[175] G. V. A. G. V. M. B. T. O. G. M. B. P. P. R. W. V. D. J. V. A. P. D. C. M. B. M. P. Fabian Pedregosa and É. Duchesnay, "Scikit-learn: Machine Learning in {P}ython," *Journal of Machine Learning Research,* vol. 12, pp. 2825--2830, 2011.

[176] " Backpropagation," 24 October 2021. [Online]. Available: https://brilliant.org/wiki/backpropagation/.

[177] W. a. K. K. a. C. C. a. G. T. a. S. J. Wang, "Machine audition: Principles, algorithms," *Information Science Reference (an imprint of IGI Global),* pp. 80-105, 2011.

[178] M. a. U. S. a. S. A. Sidorov, "Emotions are a personal thing: Towards speaker-adaptive emotion recognition," in *IEEE international conference on acoustics, speech and signal processing (ICASSP)*, Florence, 2014.

[179] X. a. G. J. a. B. R. Zhou, "Deep learning based affective model for speech emotion recognition," in *Intl IEEE Conferences on Ubiquitous Intelligence \& Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld)*, Toulouse, 2016.

[180] K. a. Y. D. a. T. I. Han, "Speech emotion recognition using deep neural network and extreme learning machine," in *Interspeech 2014*, Singapore, 2014.

[181] M. S.-G. F. M.-L. H. L. R. J. G. E. M. Albornoz, "Spoken emotion recognition using deep learning," in *Iberoamerican congress on pattern recognition*, Springer, 2014, pp. 104-111.

[182] D. a. S. M. L. a. L. J. a. H. J.-T. a. S. F. Yu, "Feature learning in deep neural networks-studies on speech recognition tasks," *arXiv preprint arXiv:1301.3605,* 2013.

[183] W. a. J. D. a. L. T. Lim, "Speech emotion recognition using convolutional and recurrent neural networks," in *Asia-Pacific signal and information processing association annual summit and conference (APSIPA)*, 2016.

[184] A. P. M. R. W. S. a. B. W. Felix Burkhardt, "A Database of German Emotional Speech," in *Interspeech*, Lisbon, 2005.

[185] S. a. B. E. a. Z. C. Mirsamadi, "Automatic speech emotion recognition using recurrent neural networks with local attention," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Toronto, 2017.

[186] D. a. Z. J. a. L. M. Tang, "An End-to-End Deep Learning Framework for Speech Emotion Recognition of Atypical Individuals," in *Interspeech*, 2018.

[187] M. a. G. P. a. P. D. a. G. N. K. a. S. K. K. a. D. N. Sarma, "Emotion Identification from Raw Speech Signals Using DNNs," in *Interspeech*, 2018.

[188] H. M. a. L. M. a. C. L. Fayek, "Towards real-time speech emotion recognition using deep neural networks," in *9th international conference on signal processing and communication systems (ICSPCS)*, Cairns, QLD, 2015.

[189] P. Sethuraman, "Towards data science," 13 September 2020. [Online]. Available: https://towardsdatascience.com/a-comparison-of-dnn-cnn-and-lstm-using-tf-keras-2191f8c77bbe. [Accessed 01 September 2021].

[190] F. Chollet, Deep Learning with Python, Manning, 2017.

[191] A. Quesada, "Data Science and Machine Learning," 2021. [Online]. Available: https://www.neuraldesigner.com/blog/5_algorithms_to_train_a_neural_network.

[192] D. P. K. a. J. Ba, "Adam: A Method for Stochastic Optimization," *arXiv preprint arXiv:1412.6980,* 2014.

[193] "Keras," [Online]. Available: https://keras.io/api/optimizers/adam/.

[194] A. Budhiraja, "Amar Budhiraja," Medium.com, 15 December 2016. [Online]. Available: https://medium.com/@amarbudhiraja/https-medium-com-amarbudhiraja-learning-less-to-learn-better-dropout-in-deep-machine-learning-74334da4bfc5.

[195] "Keras," Keras, [Online]. Available: https://keras.io/api/layers/core_layers/dense/. [Accessed 04 October 2021].

[196] E. Cambria, A. Livingstone and A. Hussain, "The Hourglass of Emotions," in *Cognitive Behavioural Systems*, Berlin, Springer Berlin Heidelberg, 2012, pp. 144--157.

[197] S. a. G. R. a. S. G. a. A. W. a. E.-W. C. ahu, "Adversarial auto-encoders for speech based emotion recognition," *arXiv preprint arXiv:1806.02146,* 2018.