Extraction of Emotional Utterances by Employing Speech Emotion Recognition and Synthesis

Thesis submitted to the Faculty of Engineering and Technology, Jadavpur University In partial fulfilment of the requirements for the degree of

Master of Engineering in Computer Science and Engineering

In the department of Computer Science and Engineering

By

Gaurab Ghosh

Examination Roll No.:	**M4CSE19018**
Class Roll No.:	**001710502010**
Registration No.:	**140749 of 2017-2018**
Session:	**2017-2019**

Under the guidance of

Dr. Dipankar Das

Department of Computer Science and Engineering

Jadavpur University, Kolkata-700 032

Faculty Council of Engineering and Technology
JADAVPUR UNIVERSITY, KOLKATA – 700032

Certificate of Recommendation

This is to certify that Gaurab Ghosh has completed his dissertation entitled "Extraction of Emotional Utterances by Employing Speech Emotion Recognition and Synthesis", under the supervision and guidance of Prof. (Dr.) Dipankar Das, Jadavpur University, Kolkata. We are satisfied with his work, which is being presented for the partial fulfilment of the degree of Master of Engineering in Computer Science & Engineering, Jadavpur University, Kolkata – 700032.

Prof. (Dr.) Dipankar Das
Faculty in Charge of Thesis

Prof. (Dr.) Mahantapas Kundu
HOD, Dept. of Computer Science & Engineering, Jadavpur University, Kolkata – 700 032

Prof. (Dr.) Chiranjib Bhattacharjee
Dean, Faculty Council of Engineering and Technology, Jadavpur University, Kolkata – 700 032

Faculty Council of Engineering and Technology
JADAVPUR UNIVERSITY, KOLKATA – 700032

Certificate of Approval[*]

The foregoing thesis is hereby approved as a creditable study of Master of Engineering in Computer Science & Engineering and presented in a manner satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion therein but approve this thesis only for the purpose for which it is submitted.

Final Examination for Evaluation of the Thesis

Signature of Examiners

[*] *Only in case the thesis is approved.*

Declaration of Originality and Compliance of Academic Ethics

I hereby declare that this thesis contains literature survey and original research work by the undersigned candidate, as part of his Master of Engineering in Computer Science & Engineering.
All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.
I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name (Block Letters):

Exam Roll Number:

Thesis Title:

Signature with date:

Acknowledgements

First, I would like to express my deepest and sincerest gratitude to my advisor, Dr. Dipankar Das, for his solicitous guidance in this research study. His knowledge and commitment have been an endless source of inspiration to me. Also I would like to thank several Open Source Courses which helped me a lot in gaining knowledge about speech processing which has been the most important part throughout this thesis.

Finally, I would like to thank all my family members and friends for their unconditional support.

Abstract

Deep learning methods have been applied to several speech processing problems in recent years. In this work we have explored different deep learning methods for speech emotion recognition and speech synthesis. We have employed normal deep feed-forward neural network (DNN) and convolutional neural network (CNN) to classify audio files according to their emotional content and we have also applied DNN model to implement a speech synthesis system from scratch.

A database consists of emotional utterances of several words has also been developed as a part of this thesis work. It contains same word in different emotional utterances. This database is ideally supposed to contain all the English words that exist in an English dictionary. But as of the writing of this thesis, the size of the database is not that large but more words can be added to this database with the corresponding utterances in the future using the proposed system.

We have also reported the results of experiments performed on the DNN and CNN models for speech emotion classification and on the DNN for Text-to-Speech synthesis system. We also reported a comparison study on the CNN and DNN models for emotion classification system.

Table of Contents

Chapter 1

Introduction

1.1 Human-Computer Interaction (HCI)

Human–Computer Interaction (HCI) researches the way we humans interact with a machine (specifically a computer) in order to improve the existing technologies. As a field of research, human–computer interaction is not any more restricted to only Computer Science domain but it is situated at the overlapping region of computer science, behavioural sciences, design, media studies, and other fields of study. The two most important aspects related to HCI research are the ways of interaction between human and computers, and the underlying technologies that let humans interact with computers in novel ways. Among the ways in which humans interact with computers, interaction through text and speech have gained a lot of interest in the research community as these are basic modes of communication for human to human interaction.

1.2 Natural Language in HCI

The need of the technology that enables computers to interpret human language has been around since the inception of computers themselves. Since natural language usually provides effortless and effective means of communication in human-human interaction, its importance and potential in human-computer interaction is overwhelming. This type of interaction which can be either spoken or written, may contribute greatly in the human-computer interaction process than other available modalities [1], such as icons, and menus, keyboard and pointing. For the users with different abilities, natural language may even be the only applicable modality to interact with a computer.

The past several decades have witnessed a wealth of studies on understanding the human brain and building systems that mimic human intelligence [2,3,4,5,6,7]. The human brain is a complicated organ that has been a lasting inspiration for research in Artificial Intelligence (AI). The neural networks of human brain are strongly competent to learn high-level abstract concepts from experiencing low-level information processed by sensory periphery. Learning language, understanding speech, and recognizing faces are some examples that manifest the remarkable power of the human brain in learning high-level concepts. The main goal of AI is to develop intelligent systems that are able to generate rational thoughts and behaviours similar to human thought and performance [8]. There are a variety of study fields that are considered as the sub-fields of AI. Computer vision, natural language processing, automated reasoning, robotics, machine listening, and machine learning are some of these major areas in AI research.

1.3 Text and Speech Modalities in HCI

Text based HCI systems have been in use since the 90s but recently due to advances in technology and computing capabilities, we can go beyond the earlier limitations,in recent trends, simple text based systems to intelligent chatbots that provides more personalized experience to the user through speech interactions.

Developing machines that interact with humans by understanding speech pave the way for building systems that are equipped with human-like intelligence. Speech is the fast and best normal way of communicating amongst human, and understanding speech is one of the most intricate processes that human brain performs. This reality motivates many researchers to consider speech signal as a quick and effective process to interact between computer and human. It means the computer should have enough knowledge to identify human voice and speech. Automatic speech recognition (ASR) has been an active field of AI research aiming to generate machines that communicate with people via speech [9, 10]. The early ASR systems mainly focused on the linguistic properties of speech to understand spoken utterances [11, 12, 13, 14].

1.4 Role of Emotions in HCI

Emotion plays an important role in our interactions with people and computers in everyday life. Emotions, some believe, are what make our interactions human. Rosalind Picard's fundamental publications on affective computing increased awareness in the HCI community regarding important roles of emotion in human-computer interactions [15, 16, 17, 18]. Since then, researchers have also become increasingly aware of the importance of emotion in the design process [19].

We are using advanced technological devices in our everyday life such as cell phones, computers, ATMs, etc. We interact with all of these devices and more by using the keyboard be it physical or virtual, that is, by using text. But if we can do the same interactive session by using our voice or our speech just like talking to a normal person expressing ourselves to the system, that would be much more fluid and seamless. By using only text for interaction we are restricting ourselves to not conversing with the system in a more one to one basis. Even though we try to personalize each and every thing we use in our daily lives, the manner in which we humans interact with the computer is not available to that level such that we can call it a truly personal device. We are not able to express our emotions when we are using text to communicate. Also text or simple text recognition might restrict us from exploring the full potential and possibilities of a modern day computing technology.

So in regard to the aforementioned discussion it can be stated conclusively that, research on the domain of speech synthesis and emotion recognition to improve the current domain of computers understanding our intentions much more clearly while interacting is truly exciting, promising and worth exploring.

1.4.1 Types of Speech Emotion

There are many different types of emotion that have an influence on how we live and interact with others. At times, it may seem like we are ruled by these emotions. The choices we make, the actions we take, and the perceptions we have are all influenced by the emotions we are experiencing at any given moment. Psychologists have also tried to identify the different types of emotions that people experience. A few different theories have emerged to categorize and explain the emotions that people feel.

Basic Emotions: During the 1970s, psychologist Paul Eckman identified six basic emotions that he suggested were universally experienced in all human cultures. The emotions he identified were happiness, sadness, disgust, fear, surprise, and anger. He later expanded his list of basic emotions to include such things as pride, shame, embarrassment, and excitement.

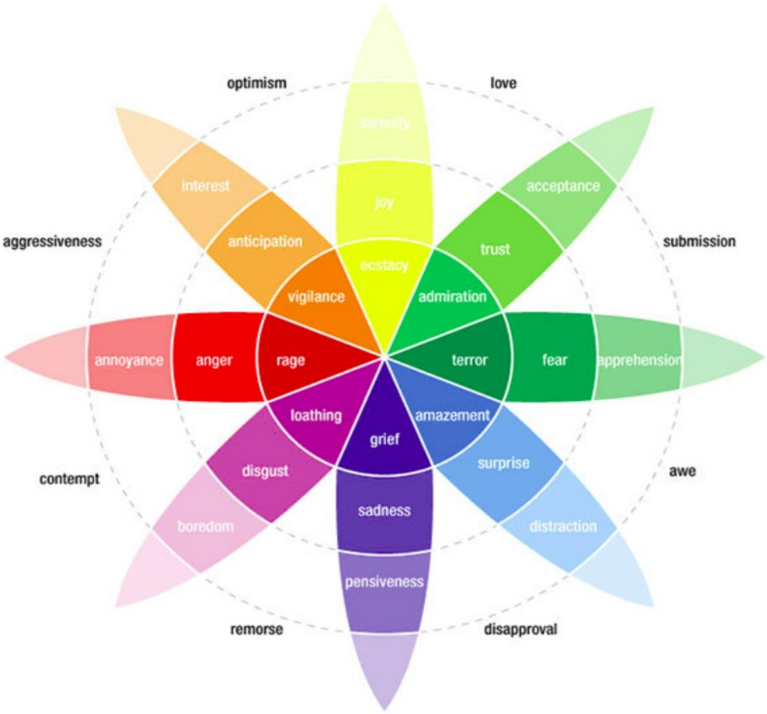


Figure 1.1*:* Plutchik's “Wheel of Emotion”

Combining Emotions: Psychologist Robert Plutchik put forth a "wheel of emotions" that worked something like the color wheel. Emotions can be combined to form different feelings, much like colors can be mixed to create other shades. According to this theory, the more basic emotions act something like building blocks. More complex, sometimes mixed emotions, are blendings of these more basic ones. For example, basic emotions such as ‘joy’ and ‘trust’ can be combined to create ‘love’.

1.5 Applications of Speech Emotion Recognition(SER)

Speech emotion recognition is mostly beneficial for applications, which need human-computer interaction such as speech synthesis, customer service, education, forensics and medical analysis. Emotion Recognition is used in call center for classifying calls according to emotions [20]. Emotion Recognition serves as the performance parameter for conversational analysis [21] thus identifying the unsatisfied customer, customer satisfaction so on. SER is used in-car board system based on information of the mental state of the driver can be provided to the system to initiate his/her safety preventing accidents to happen [22].

1.5.1 Affective Computing

Affective computing and Human Computer Interaction (HCI) research target four broad areas: a) reducing user frustration, b) comfortable communication of user emotion, c) infrastructure and applications to process affective information and d) building tools to support development of socio-emotional skills [23]. Without information about emotions, it is difficult to achieve a harmonic and natural man-machine interface for applications such as patient care, geriatric nursing, call centres, psychological consultation, and human communication[24].

1.5.2 Emotion recognition in Health Care

Health Care industry is among many other industries which leverages emotion recognition techniques to solve complex patient related problems and to improve the life of the patients. One of the many applications of the emotion recognition is to decide when patients need medicine based on his/ her emotional state or to help out physicians determine whom to see first. Another important aspect of emotional recognition in health care is to detect depression, Post Traumatic Stress Disorder (PTSD) and suicidal tendencies to prevent these as much as possible. These mental conditions are seen majorly in both military and civilian situations. These mental diseases almost always combine with emotional changes. So, if we could detect those emotional changes expressed by a patient in a normal conversation, using the Emotion recognition technologies, then it would greatly help our community.

1.5.3 Automotive industry and Emotion Recognition

Emotion detection and regulation have become an important aspect of research in automotive industry. Making cars safer and more personal is something that drives the car manufacturers around the world and for this reason they are very much interested to focus on Emotion recognition technologies. As driving a car is a continuous process, recognition of emotion at a specific time is not sufficient. It needs to constantly monitor the driver as different emotional states heavily influence driving performance. Whenever a driver is driving recklessly or feeling drowsy, the speech and facial emotion detection system installed in the car can instantly alert the driver and passengers inside the car to take appropriate action. The smart car can also alert its neighbouring traffics based on the output of the recognition software, thus reducing the probability of a collision.

1.6 Motivation

Speech is an information-rich signal that contains paralinguistic information as well as linguistic information. As a result of this speech conveys more emotional informations than text. This reality motivates many researchers to consider speech signal as a quick, effective and natural process to interact between computer and human. It means the computer should have enough knowledge to identify human voice and speech and the underlying emotion for naturally interact with humans.

This also motivated us to crate a system to recognize emotions from spoken signals and we implemented several models to do the task. To the best of our knowledge, there exists no emotional-word level database as of writing this thesis, we were very much motivated to develop one using the proposed emotion detection system and word segmentation method from audio files.

Natural sounding speech generation is also an important aspect of NLP research as the Emotion Recognition. This greatly influences the interaction between a human and a computer. As we have already established that emotion plays an important role in Human-Computer Interaction, generation of natural sounding speech is therefore very much required.

1.7 Challenges

1.7.1 Speech Emotion Recognition

Although, there is a significant improvement in speech recognition but still researchers are away from natural interplay between computer and human, since computer is not capable of understanding human emotional state. The recognition of emotional speech aims to recognize the emotional condition of individual utterer by applying his/her voice automatically. Recognizing of emotional conditions in speech signals are so challenging area for several reasons.

- First issue of all speech emotional methods is to select the best features, which are powerful enough to distinguish between different emotions.
- The presence of various languages, accents, sentences, speaking styles, speakers also adds another difficulty because these characteristics directly change most of the extracted features including pitch, energy [2].
- Furthermore, it is possible to have a more than one specific emotion at a time in the same speech signal, each emotion correlate with a

different part of speech signals. Therefore, defining the boundaries between parts of emotion is very challenging task.

1.7.2 Speech Synthesis

The problem area in speech synthesis is very wide. There are several problems in text pre-processing, such as numerals, abbreviations, and acronyms. Correct prosody and pronunciation analysis from written text is also a major problem today.

Written text contains no explicit emotions and pronunciation of proper and foreign names is sometimes very anomalous. At the low-level synthesis, the discontinuities and contextual effects in wave concatenation methods are the most problematic.

Speech synthesis has been found also more difficult with female and child voices. Female voice has a pitch almost twice as high as in contrast to male voice and with children it may be even three times as high. The higher fundamental frequency makes it more difficult to estimate the formant frequency locations (Klatt 1987, Klatt et al. 1990).

The evaluation and assessment of synthesized speech is neither a simple task. Speech quality is a multidimensional term and the evaluation method must be chosen carefully to achieve desired results. This chapter describes the major problems in text-to-speech research.

1.8 Hypotheses

1.8.1 Speech Emotion Recognition

- A single word can be associated with multiple emotions [25]. Based on this hypothesis we have built our emotion classifier and chosen datasets carefully.
- Although there are several other modalities such as facial expression, body language, through which emotions can be expressed but we limited our study to speech modality and have not considered other modalities. As a result, the databases that were acted by actors were used in the current study because the

emotions were expressed with exaggeration by actors, which potentially compensates for the lack of information provided by in other modalities. This allows us to explore the effectiveness of deep learning models with greater control compared with using daily-life utterances.

- We limited our model to classify emotions for English language only.

1.8.2 Speech Synthesis

- As it is difficult to simultaneously generate both male and female voices in a text-to-speech system, we limited ourselves to synthesize only male voice. For this reason we chose our dataset with only male speakers.

1.9 Contributions

1.9.1 Speech Emotion Recognition

We have proposed two systems based on Deep Learning method to classify a speech signal according to its emotional content.

1. The first model is based on simple Deep Feed-forward Neural Network. As it is a very basic model, it was unable to recognize enough important features from speech signal to correctly classify it. The overall accuracy we achieved from this system is approximately 40%.
2. The second model we implemented, is based Convolutional Neural Network model. Our main contribution lies in the way we applied the CNN model to our dataset. In many studies CNN have been used to classify speech emotion but the CNN model was applied on the spectrogram image which is a visual representation of the spectrum of frequencies of an audio signal. In contrast, we have applied our CNN model on the array of low-level MFCC features, extracted from the

spectrogram image of an audio signal. Due to this fact we used 1-Dimensional Convolutional layers in our CNN and not 2-Dimensional ones, which are generally used on image data. The overall accuracy we achieved from this model is approximately 65%.

Now apart from these two systems, we have also developed a Database which has been discussed in details in Chapter 4. This database contains several words and their emotional class and the utterance of that particular word in one or more categories of emotion (Same word can belong to multiple categories of emotion). As of the writing of this thesis, any database of this kind has not existed to the best of our knowledge. Thus, this is one of the main contributions of our work in the betterment of Speech Emotion Research.

1.9.2 Speech Synthesis

We have reported a very basic system using Deep Feed-forward Neural Networks architecture to convert any written text to speech. This work is performed only to gain insight and knowledge about the workings of a Text-to-speech system. This also helps in generating all types of speech level utterances of both emotional and non-emotional words either including or excluding their emotional slants with respect to each of the corresponding text words.

1.10 Thesis Outline

The first chapter of the thesis discusses the importance of Natural Language Processing in Human Computer Interaction and also the importance of emotion in HCI systems. It also describes several application of speech emotion recognition system in many real-world problems. It also discusses the main contributions of our work in a very brief manner.

The second chapter details the current research progress in the speech processing and speech emotion recognition areas. It contains a very informative table about several research works like what databases they have used and what classifiers work better than other.

The third chapter contains detailed discussion on the two speech emotion classifier that we have developed for our study and the datasets we used. The first one is the DNN model and second one is the CNN model. This chapter also gives a background study and working principles of the two models. It also discusses the results of the two classifiers.

The fourth chapter discusses in detail the method for developing the Dataset which contains the emotion-tagged words and their corresponding speech files.

Fifth chapter gives a brief overview of the mechanism of a basic Text-to-Speech system. It also reports the results we obtained after training our basic DNN model for speech synthesis task.

Chapter 2

Literature Survey

2.1 Main Aspects of Speech Emotion Research

If we observe a comprehensive review of speech emotion recognition systems targeting pattern recognition researchers who do not necessarily have a dee p background in speech analysis, we notice three main aspects of this research field:

(1) Important design criteria of emotional speech corpora,

(2) The impact of speech features on the classification performance of speech emotion recognition.

(3) Classification systems employed in speech emotion recognition.

Different types of features and the benefits of combining the available acoustic information with other sources such as linguistic, discourse, and video have been surveyed. Different classification techniques commonly used in speech emotion recognition have been studied. Numerous speech recognition systems implemented in other research papers are included in order to have an insight on the performance of existing speech emotion recognizers.

2.1.1 Design Criteria of Emotional Corpora

According to some study there are certain criteria that are used to measure how a specific emotional corpora simulates real-world emotional expressions. The following are the most important factors among others:

Real-world emotions or acted ones:

It has always been criticized that acted emotions are not the same as real ones. Acted emotions tend to be more exaggerated than real ones. Nonetheless, the relationship between the acoustic correlate and the acted emotions does not contradict that between acoustic correlates and real ones.

Distribution of utterances over emotions:

Some corpus developers prefer that the number of utterances across each emotion is almost the same in order to properly evaluate the classification accuracy such as in the Berlin corpus [26]. Alternatively, other researchers prefer that the number of utterances should reflect their real-world frequency.

Same statement with different emotions:

It is common in many corpora to record the same sentence with different emotions to study the explicit effect of emotions on the acoustic features of the speech utterances,. One advantage of such a database is to ensure that the human judgment on the perceived emotion is solely based on the emotional content of the sentence and not on its lexical content.

2.1.2 Speech Features

Extraction of suitable features has been one of the most important factors since the inception of any classification system. Speech emotion classification system is no exception. Extraction of relevant speech features is very crucial to efficiently characterize different emotions.

There are mainly three aspects of feature extraction that should be discussed to have a better idea about speech emotion classification system.

1. The region of analysis used for feature extraction. Based on this we can differentiate between local features and global features.
2. The effect of pre-processing audio or silence removal or post-filtering of audio on the overall performance of the classifier.

3. Whether the acoustic features are sufficient for modeling the emotions or combination of linguistic or discourse information with the acoustic features is necessary.

2.1.2.1 Local V/S Global Features

Since speech signals are not stationary even in wide sense, it is common practice in speech processing to divide a speech signal into small segments called frames. Within each frame the signal is considered to be approximately stationary. Prosodic speech features such as pitch and energy are extracted from each frame and called local features. On the other hand, global features are calculated as statistics of all speech features extracted from an utterance. Some of the most used global features in SER are Fundamental Frequency (F_0), Energy, Formants etc.

2.1.2.2 Continuous speech features

Continuous speech features such as pitch and energy are heavily used in speech emotion recognition as most researchers believe that these features convey much of the emotional content of an utterance. In a study carried out by Banse et al.[27], they examined vocal cues for 14 emotion categories. The speech features that have been used in this study are related to the fundamental frequency (F_0), the energy, the articulation rate, and the spectral information in voiced and unvoiced portions.

According to many studies, these acoustic features can be grouped into the following categories:

(1) Pitch-related features;

(2) Formants features;

(3) Energy-related features;

(4) Timing features;

(5) Articulation features.

2.1.2.3 Spectral-based speech features

Along side with the continuous speech acoustic features, spectral features are also often used as a short-time representation for speech signal. According to the literature, the distribution of the spectral energy across the speech range of frequency greatly affects the emotional content of an utterance. Spectral features can be extracted in a number of ways including the ordinary linear predictor coefficients (LPC), one-sided autocorrelation linear predictor coefficients (OSALPC), short-time coherence method (SMC), and least-squares modified Yule–Walker equations (LSMYWE). However, in order to better exploit the spectral distribution over the audible frequency range, the estimated spectrum is often passed through a bank of band-pass filters. Spectral features are then extracted from the outputs of these filters.

Cepstral based features can be derived from the corresponding linear features as in the case of linear predictor cepstral coefficients (LPCC) and cepstral based OSALPC (OSALPCC). In [28], the authors have shown that cepstral based features such as LPCC, OSALPCC and Mel-frequency cepstrum coefficients (MFCC) clearly outperform the performance of LPC and OSALPC for detection of stress in speech signal.

2.2 Different Types of Features, Classifiers and Datasets in Recent Speech Emotional Recognition Systems

The below table gave us a brief but concise overview of the recent trend in the field of speech emotion recognition research. The table contains 6 columns namely, Reference, Type of classifier, Recognition Rate, Type of Dataset and Methods.

Reference represents the literature in which the corresponding study was carried out. Type of Classifier represents the name of the classifier used in that particular study. Recognition Rate conveys the accuracy of the used classifier. Type of Dataset represents the name of the datasets used in the study. Methods column gives us a detailed understanding of the classifier used in the study.

Refernce	Types of classifier	Types of features	Recogniti on Rate	Type of Dataset	Methods
H. Cao et al. 2015 [29]	SVM	Prosodic and spectral features	44.4%	Berlin & LDC & FAU Aibo dataset	Ranking SVM
L. Chen et al.2012 [30]	SVM & ANN	Energy, ZCR, pitch, SC, spectrum cut-off frequency, correlation density (Cd), fractal dimension, MFF.	86.5%, 68.5% and 50.2% for different level	Beihang University Database of Emotional Speech (BHUDES)	Multi-level SVM classifier & ANN to reduce dimensionality
T. L. Nwe et al.2003 [31]	HMM	Log frequency power coefficients (LFPC), MFCC	Average and best result 78% and 96% respectivey	Two private speech dataset	Discrete HMM and LFPC to characterize speech signal.
S. Wu et al.2011 [32]	SVM & RBF	Modulation spectral features (MSFs)	91.6%	Berlin & VAM	Modulation filterbank & auditory filterbank for speech decomposition, SVM & RBF for classification.
J. Rong et al. 2009 [33]	Decision tree & random forest	Linguistic , spectral-related , contour related, tone-based and /or vowel-related features	Best 82.54% & worst 16%	Private series of Chinese emotional speech dataset	Ensemble random forest to trees (ERFTrees) method with a high number of features.
C. H. Wu	GMM,	Prosodic	MDT 80%,	Private	Fusion method

et al.2011 [34]	SVM, MLP and MDT(Meta decision tree)	information and semantic labels (SL), acoustic and prosodic features	SLbased 80.92%, mixture of AP & SL 83.55%	database	based on AP & SL and multiple classifier by maximum entropy model (MaxEnt).
S. S. Narayanan 2005 [35]	k-NN & linear discriminate	Fundamental frequency (F0), energy, duration, and the first and formant	40.7% for males & 36.4% for females	Private speech database from call center	Domain-specific emotion recognition by k-NN and linear discriminate classifier
B. Yang et al.2010 [36]	Bayesian learning framework	Energy, pitch statistics, duration, formant, and zero-crossing rate (ZCR	Best for sadness 87.6% and overall 2% improvemt	Berlin emotion dataset	Harmony features with Bayesian classifier with Gaussian class conditional likelihood
E. M. Albornoz et al.2011 [37]	HMM, GMM, MLP and hierarchical model	Mean of the log-spectrum (MLS), MFCCs and prosodic features	HMM 68.57, Hierarchical model 71.75	Berlin dataset	Spectral characteristics of signals are used in order to group emotions based on acoustic rather than psychological considerations.
J. H. Yeh et al. 2011 [38]	k-NN	Jitter, shimmer, formants, LPC, LPCC, MFCC, LFPC, PLP, and Rasta-PLP, SFS and SBS	Best 86%	Chinese emotional speech corpus We invited 18 males and 16 females.	Segment based method by employing k-NN, SFS (sequential forward selection), SBS(sequential backward selection)

M. Grimm et al. 2007 [39]	k-NN	Acoustic features such as pitch, energy, speaking rate and spectral characteristics	83.5%	EMA and VAM dataset	A multi-dimensional model by utilizing emotion primitives. Three dimension were made by composing of three different value of emotion primitives, which is called valence, activation, and dominance.
J. P. Arias et al. 2014 [40]	Binary classifier & QDC	Prosodic features such as energy contour and duration	75.8%	SEMAINE databases	Shape based method by using functional data analysis to obtain natural changeability.

Table 2.1: Review of the studies conducted recently on Speech Emotion Recognition

Chapter 3

Speech Emotion Classification

3.1 Databases

3.1.1 SAVEE: British English Database

The **Surrey Audio-Visual Expressed Emotion** (SAVEE) database was recorded from four native English male speakers (identified as DC, JE, JK, KL), postgraduate students and researchers at the University of Surrey aged from 27 to 31 years. Emotion has been described psychologically in discrete categories: anger, disgust, fear, happiness, sadness and surprise [41]. This is supported by the cross-cultural studies of Ekman [42] and studies of automatic emotion recognition tended to focus on recognizing these [43]. We added neutral to provide recordings of 7 emotion categories. The text material consisted of 15 TIMIT sentences per emotion: 3 common among all emotions, 2 emotion-specific and 10 generic sentences that were different for each emotion and phonetically-balanced. The sampling rate of all recordings was 44.1 kHz. The 3 common and $2 \times 6 = 12$ emotion-specific sentences were recorded as neutral to give 30 neutral sentences. This resulted in a total of 120 utterances per speaker, for example:

Common: She had your dark suit in greasy wash water all year.

Anger: Who authorized the unlimited expense account?

Disgust: Please take this dirty table cloth to the cleaners for me.

Fear: Call an ambulance for medical assistance.

Happiness: Those musicians harmonize marvelously.

Sadness: The prospect of cutting back spending is an unpleasant one for any governor.

Surprise: The carpet cleaners shampooed our oriental rug.

Neutral: The best way to learn is to solve extra problems.

3.1.2 RAVDESS: Emotional Speech and Song Database

The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) [44] contains 7356 files (total size: 24.8 GB). The database contains 24 professional actors (12 female, 12 male), vocalizing two lexically-matched statements in a neutral North American accent. Speech includes calm, happy, sad, angry, fearful, surprise, and disgust expressions, and song contains calm, happy, sad, angry, and fearful emotions. The statements are “Kids are talking by the door” and “Dogs are sitting by the door”. Each expression is produced at two levels of emotional intensity (normal, strong), with an additional neutral expression. All conditions are available in three modality formats: Audio-only (16bit, 48kHz .wav), Audio-Video (720p H.264, AAC 48kHz, .mp4), and Video-only (no sound).

We used only the audio modality as our focus was on the recognition of emotion from speech. Speech file (Audio_Speech_Actors_01-24.zip, 215 MB) contains 1440 files: 60 trials per actor x 24 actors = 1440.

3.1.3 Data Cleaning and Pre-processing

In order to have a consistent sampling rate across all databases, all utterances were resampled and filtered by an antialiasing FIR lowpass filter to have frequency rate of 44.1 kHz prior to any processing. All audio utterances were then converted into spectrograms. A spectrogram is an image that displays the variation of energy at different frequencies across time. The vertical axis (ordinate) represents frequency and the horizontal axis (abscissa) represents time. The energy or intensity is encoded either by the level of darkness or by the colors. There are two general types of spectrograms: wide-band spectrograms and narrow-band spectrograms. Wide-band spectrograms has a higher time resolution than narrow-band spectrograms. This property enables the wide-band spectrograms to show individual glottal pulses. In contrast, narrow-band spectrograms have higher frequency resolution than wide-band spectrograms. This feature enables the narrow-band spectrograms to resolve individual harmonics [66, 67]. Figure 4.1 depicts the wide-band and narrow-band spectrogram images of a speech utterance. Considering the importance of vocal fold vibration, along with the fact that glottal pulse is associated with one period of vocal fold vibration [66], we decided to convert all utterances into wide-band spectrograms.

3.2 Model 1: Deep Neural Network Model

3.2.1 Neural Networks

A Neural network consists of a huge number of simple processing elements called neurons or nodes. Each of the neurons in a layer is connected with every neurons in its preceding layer and each of this association carries a value which is called weight of that particular link or association. Neural networks employ a special type of algorithms, which can loosely mimic the human brain. These type of networks are applied to a wide variety of problems specifically related to classifying patterns, performing mapping from input to output, grouping similar patterns. In order to enable the Neural Networks to identify patterns, all types of input data, be it images, sound, text or time series, must be translated into vectors containing numerical values.

Neural networks can help us mainly in clustering and classification problems. Clustering problems generally refers to grouping of unlabelled data according to the similarities among the input data, and on the other hand in classification problems a test data is classified based on the trained model which was previously trained using a labelled dataset. These types of networks can also be used as a front end to select features that are in turn fed to other clustering or classification algorithms. This greatly helps advanced machine learning applications such as supervised learning, unsupervised learning. In what follows, we give a brief introduction on supervised learning as it has been applied in the current study.

In supervised learning, the training data incorporate the desired response, called labels; that is, for each observation (training sample or instance), there is a corresponding label. The goal of learning is to predict the label of each training/test instance as correctly as possible. Classification is one example of supervised learning wherein the machine learning algorithm learns to classify the input data into two or more categories. To do so, the algorithm learns the discriminant features or attributes across different categories or classes based on observing training data. These features are later used to classify new test input data. Artificial neural networks (ANNs) are one well-known instance of supervised learning algorithms although some ANNs can be trained by unsupervised learning [45]. ANNs are inspired by the way biological neural networks, such as human central nervous system, work. That is, they consist of a highly interconnected processing units, called neurons. ANNs are the basic building blocks of deep learning, which is a strong modern machine learning paradigm. In fact, deep learning models are ANNs with a plethora of neurons and layers. Deep learning models have achieved remarkable successes in various machine learning applications such as classification. The key concept that makes deep learning models efficacious is their ability to learn complex features out of simple features [46]. That is, the first layers of deep learning models represent simple and basic features of the training data. The deeper layers build a complex representation by using these low-level features. This ability of building high-level features out of low-level features can potentially reduce the amount of preprocessing required for extracting hand-crafted features before designing classifiers.

3.2.2 Multi-layer Perceptron Networks

The network architecture is one of the factors that characterizes artificial neural networks (ANNs). It determines the way neurons, the basic processing units, are connected to one another. The multi-layer perceptron (MLP) is a long-established ANN architecture that is composed of neurons called linear threshold units (LTUs) [38]. Figure 3.1 illustrates a linear threshold unit. As demonstrated, an LTU receives weighted inputs from different neurons (here from three neurons) and computes the linear combination of the inputs as $z = w_1 x_1 + w_2 x_2 + w_3 x_3 + b$, where b is the bias term. Then, a step function (e.g., Heaviside function H(x) or Sign function *sgn(x))* is applied on the linear combination to generate the output as $y = f\ (z)$ where f is a step function. If the weighted sum is greater than a threshold value (which is affected by the bias term), the LTU will generate an output.

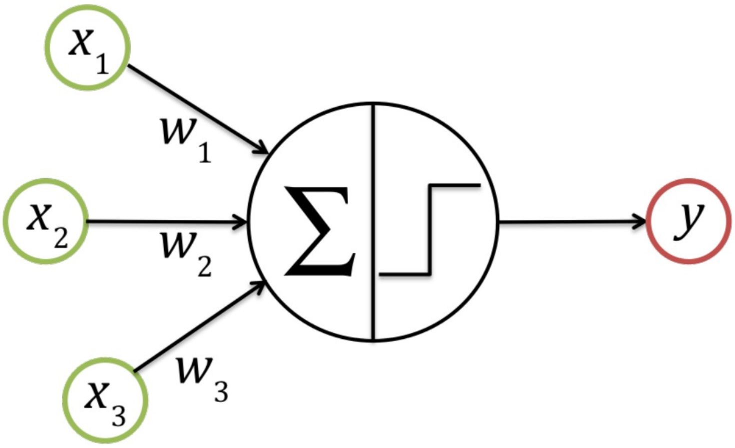


Figure 3.1: Linear threshold unit. It processes the inputs by computing the linear combination of the inputs and applying a step function.

Generally, an MLP incorporates one input layer, one or more LTU layers (called hidden layers), and one output layer. The information flows from the input layer (lower level) toward the output layer (higher level). That is why they are called feedforward artificial neural networks. The input layer represents the values of one training sample in different dimensions. This can be the amplitude of an audio signal at different sampling points or the intensity of an image at different pixels. The input is usually denoted as a vector $\vec{x}$ whose length indicates the number of dimensions (e.g., the number of sampling points in an audio signal or the number of pixels in an image). The output is either a real number, y, or a vector, $\vec{y}$, which shows the label of the input. An MLP is mainly configured as the layers of the neurons denoted as $l^{[0]}$, $l^{[1]}$, $l^{[2]}$, . . . , $l^{[n]}$, $l^{[n+1]}$, where $l^{[0]}$ is the input layer, $l^{[1]}$, $l^{[2]}$, . . . , $l^{[n]}$ are the n hidden (LTU) layers, and l [n+1] is the output layer. Every neuron within the layer $l^{[j]}_{1\leq j\leq(n+1)}$ (all layers except the input layer) directly receives the weighted input from every neuron within a layer that is one level low,that is, $l^{[j-1]}$. Figure 3.2 demonstrates an MLP with one hidden layer where W_1 and W_2

are matrices associated with the values that weight the inputs of the neurons in the hidden layer and the output layer, respectively. The vectors $\vec{b}_1$ and $\vec{b}_2$ are the weights associated with bias terms.

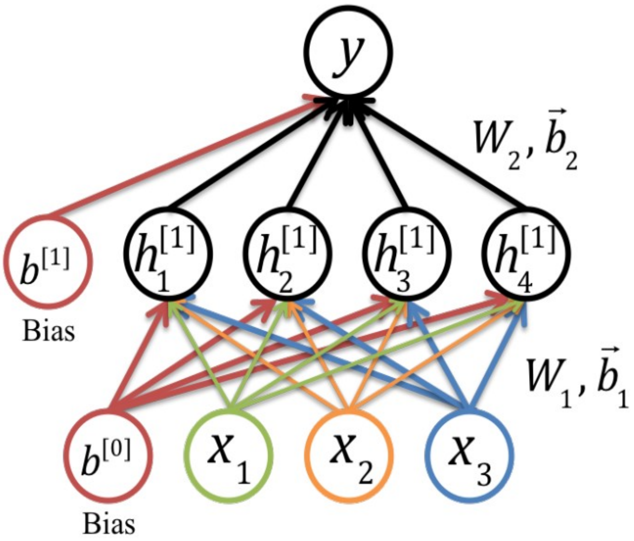


Figure 3.2: Multi-layer Perceptron. W_1 and $\vec{b}_1$ denote the weight matrix and the bias vector associated with the input layer. W_2 and $\vec{b}_2$ are the weight matrix and the bias vector associated with the hidden layer.

In fact, the weight matrices (W_1 and W_2) and the bias vectors ($\vec{b}_1$ and $\vec{b}_2$) are the parameters of interest. That is, the network learns to classify the input data by adjusting the values of these parameters. There are several learning techniques that can find the optimum values of these parameters. Backpropagation is one example of these learning methods that has been widely used to train MLP networks. The basis of the backpropagation algorithm is gradient descent, which is briefly introduced in the next section.

3.2.3 Gradient Descent

Searching for the optimum values of the weight parameters can be viewed as an optimization problem. The main objective to actually introduce the error function or loss function is to find the values of important parameters such that it minimizes the error or loss function.

There are several ways to define the error function. Equation 3.1 displays a well-established error function called sum of squared errors [47] where W, *d*, and D stand for weights, one training instance, and all training data, respectively. As shown, the overall error value (*E*) is the summation of the squared value of error (difference) between the actual label $(\vec{t_d})$ and the predicted label $(\vec{y_d})$ over all training data (*D*). The objective is to find the optimal weight space for the values that minimize this function.

$$E(W)=\frac{1}{2}\sum_{d\in D}(\vec{t_d}-\vec{y_d})^2 \quad . \tag{3.1}$$

Gradient descent is a common optimization algorithm that is used to minimize the error function. Suppose we have a linear unit that computes the weighted sum of its inputs as follows:

$$z = w_1\ x_1 + w_2\ x_2 + ... + w_p\ x_p + b = \vec{w}\cdot\vec{x} + b \tag{3.2}$$

where b, $\vec{w}$, $\vec{x}$ and z are the bias term, weight vector, the input vector, and the output of the linear unit, respectively. This unit's error function is

$$E(\vec{w})=\frac{1}{2}\sum_{d\in D}(\vec{t_d}-\vec{y_d})^2 \quad . \tag{3.3}$$

Gradient descent, which is an iterative technique, begins with a random initial values of the weight parameters and updates the weights in the opposite direction of the gradient of the error function as presented in equation 3.4 [47]

$$w_i=w_i-\eta\frac{\partial E}{\partial w_i} \quad , \tag{3.4}$$

where w_i is the weight associated with the i^{th} dimension of the input and η is the learning rate that determines the step of updates.

The back-propagation algorithm employs gradient descent to find the local minima of the error function of the Multi-Layer Perceptron (MLP) networks. The error function of the MLP networks is not convex as it is in a linear unit. As a result, finding global minima is not guaranteed. Since we do not have access to the output of the hidden neurons, the backpropagation algorithm takes advantage of the chain rule of calculus and computes the contribution of hidden neurons to the output error to update the weights associated with the hidden layers [46, 45]. It should be noted that the step function of the MLP networks poses challenges for taking the derivative with respect to the weights. Therefore, the step function is replaced with the sigmoid function, $\sigma(x)=\frac{1}{1+e^{-x}}$,which is differentiable. For more details, we refer the reader to [47,46,45].

3.2.4 Framework

Deep feed-forward neural network constitutes several layers of hidden neurons, where each neuron is connected to every neuron in its previous layer. The first layer is called input layer. For our study, the input layer consists of 216 MFCC features extracted from the audio data. The batch size has been set to 16. Thus the dimension of our input data is *(16 X 216)*. We employed three hidden layers in our architecture as depicted in Figure 2. The number of neurons in the first, second and third hidden layer are 256, 512 and 256 respectively. We have used the Rectified Linear Unit (*ReLU)* activation function in all of the three hidden layers to achieve non-linearity. Only in the output layer, *softmax* activation function is used as it gives the probability distribution across 10 output classes. As the problem is a classification problem, we have used the Cross-entropy loss. Adam optimizer is also employed to minimize the loss function across the training data. We have also employed a dropout rate of 20% after every hidden layer. The dropout layers are employed to reduce the over-fitting problem.

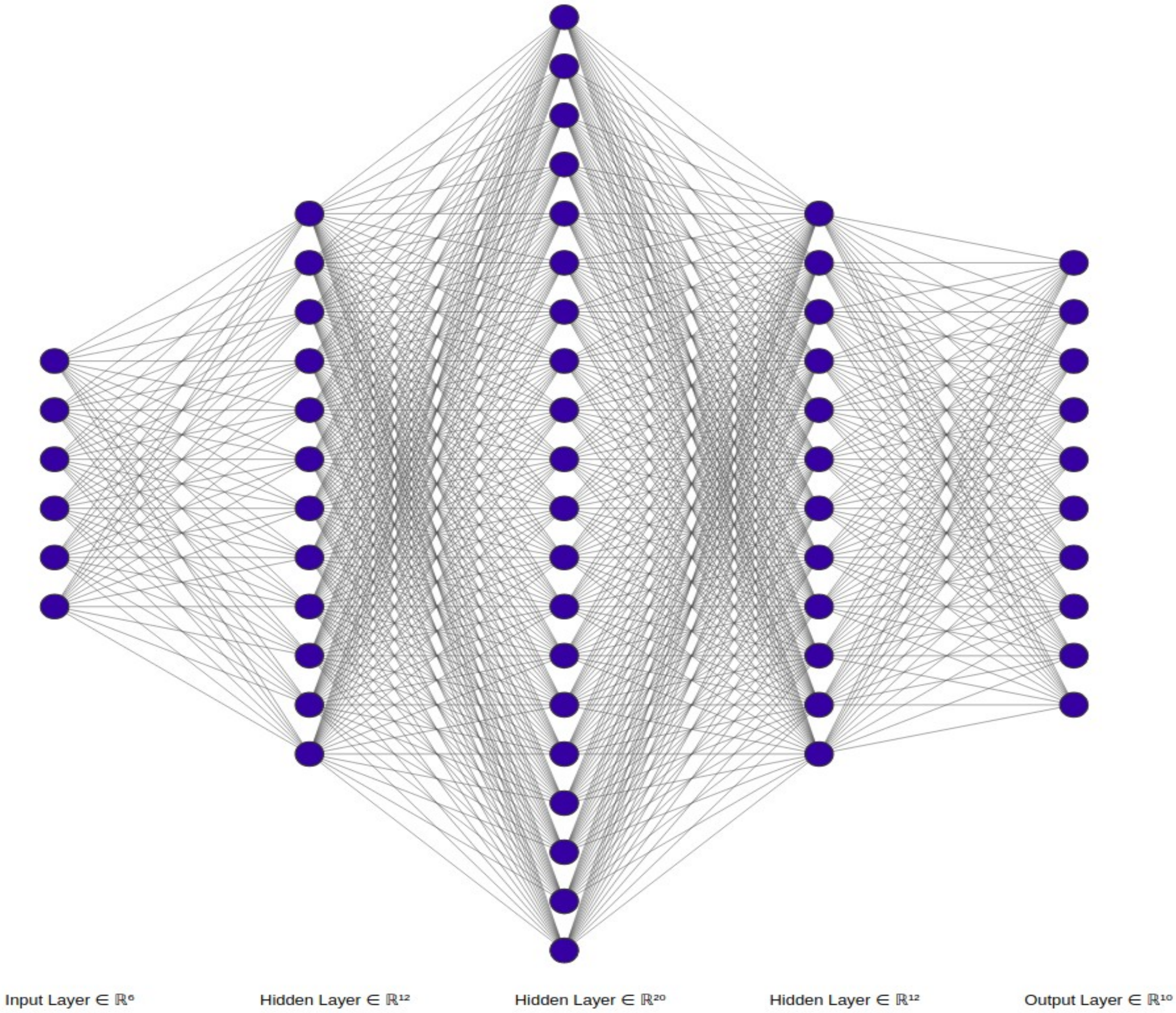


Figure 3.3: Baseline framework of Feed-forward neural network

3.3 Model 2: Convolutional Neural Network Model

Convolutional neural networks (CNNs) are one of the most popular deep learning models that have manifested remarkable success in the research areas such as object recognition [48], face recognition [49], handwriting recognition [50], speech recognition [51], and natural language processing [52]. The term convolution comes from the fact that convolution, the mathematical operation, is employed in these networks. Generally, CNNs have three fundamental building blocks: i) convolutional layer, ii) pooling layer, and iii) fully connected layer. Following, we describe these building blocks along with some basic concepts such as softmax unit, rectified linear unit, and dropout.

In the following section we described the main working principle of a CNN with respect to an image. In later sections we also described how we can apply CNN to NLP tasks.

3.3.1 Convolutional Layer

One of the main differences between the frameworks of a normal feed-forward deep neural network and Convolutional Neural Network is that the CNNs have Convolutional layers which use a mathematical fuction known as convolution instead of multiplication to compute the output. As a result, the neurons in the convolutional layers are not connected to all the neurons in their preceding layers. This architecture is inspired by the fact that neurons of the visual cortex have local receptive field [46, 45]. That is, the neurons are specialized to respond to the stimuli limited to a specific location and structure. As a result, using convolution introduces sparse connectivity and parameter sharing to CNNs, which decreases the number of parameters in deep neural networks drastically.

Figure 3.4 demonstrates the convolution of a kernel, which is a 2 × 2 matrix, with a one-channel 3 × 3 image. The output is a volume of 2 × 2 × 1. Generally, the size of output is $(n_h - f + 1) \times (n_w - f + 1) \times n_f$, where n_h is the height of the input, n_w is the width of the input, and n_f is the number of kernels. The depth of the kernel is determined by the depth of the input.

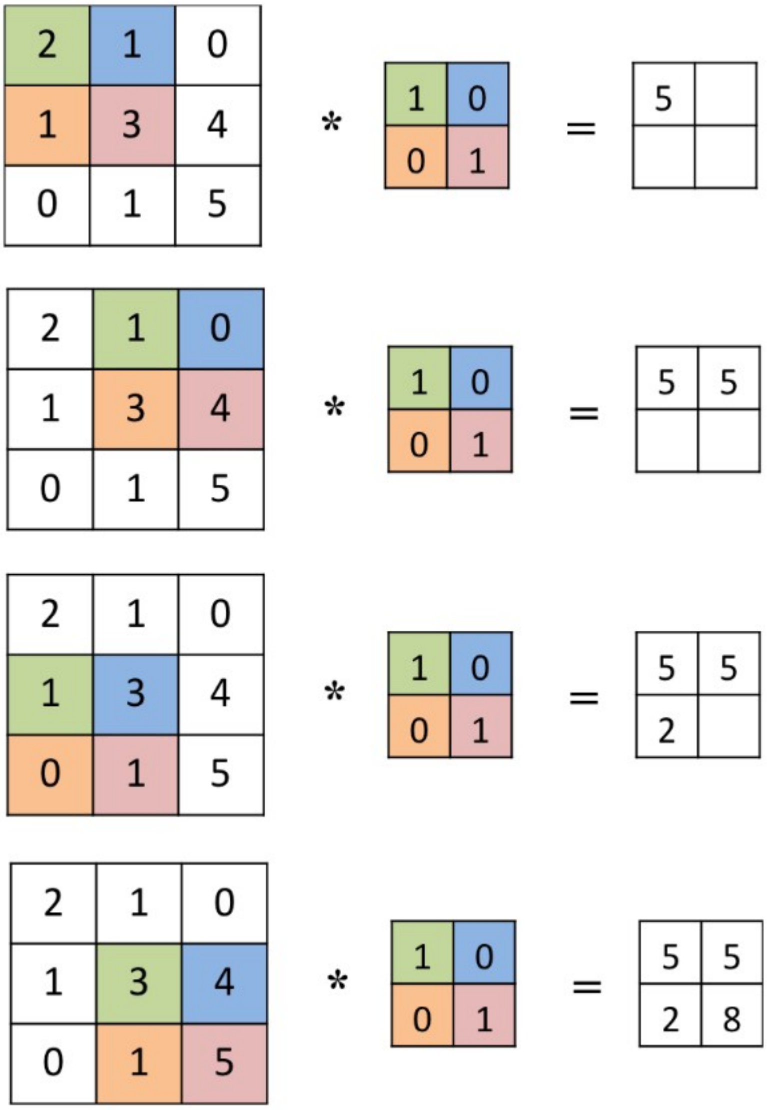


Figure 3.4: The convolution of a 3 × 3 image by a 2 × 2 kernel with a stride of 1.

For the example demonstrated in Figure 3.4, the depth of the input is $n_c = 1$. As a result the depth of kernel is 1. Also, the depth of the output is 1 since there is only one kernel. As can be seen, each output neuron is the weighted sum of the input neurons within the corresponding receptive field, which introduces sparse connectivity in CNNs. Further, the kernel is shared across the layer, which introduces parameter sharing in CNNs. The step by which the kernel slides along the input is called stride. In our example

(Figure 3.4), the stride is $s = 1$, which means that the kernel shifts one step over the image. It should be noted that the input volume shrinks after each convolutional layer. To avoid this decrement, we can pad the outer edge of the input with zero.

The local filtering that happens in convolutional layers allows detecting different basic low-level features of interest and generating various feature maps. The deep layers use these feature maps to construct the high-level representation of the inputs.

3.3.2 CNN Applied to NLP

Instead of image pixels, the input to most NLP tasks are sentences or documents or speech signals represented as a matrix. Each row of the matrix corresponds to one token or one speech file, typically a word, or an array of MFCC features. That is, each row is vector that represents a word or a speech signal. Typically, these vectors are *word embeddings* (low-dimensional representations) like word2vec or GloVe for text related work and an array of MFCC feature values for speech processing work. For a 10 word sentence using a 100-dimensional embedding we would have a 10×100 matrix as our input. That's our "image".

Here we have taken an example of applying CNN for text processing.

In vision, our filters slide over local patches of an image, but in NLP we typically use filters that slide over full rows of the matrix (words). Hence, the "width" of our filters is usually the same as the width of the input matrix. The height, or *region size*, may vary, but sliding windows over 2-5 words at a time is typical. A Convolutional Neural Network applied to NLP task look like the network depicted in Figure 3.5 below.

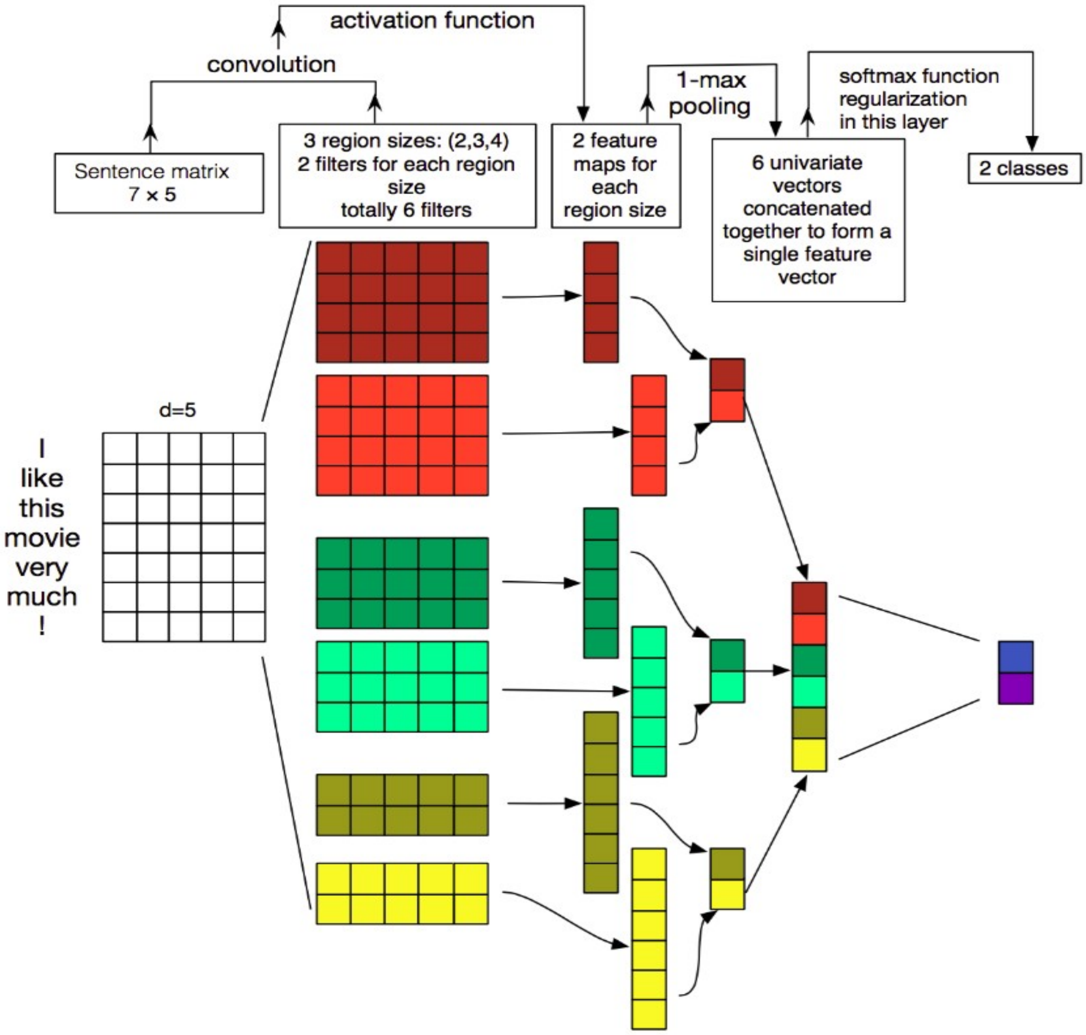

activation function
convolution
1-max pooling
softmax function regularization in this layer
Sentence matrix 7 × 5
3 region sizes: (2,3,4) 2 filters for each region size totally 6 filters
2 feature maps for each region size
6 univariate vectors concatenated together to form a single feature vector
2 classes
d=5
I
like
this
movie
very
much
!

Figure 3.5: Illustration of a Convolutional Neural Network (CNN) architecture for sentence classification. Here we depict three filter region sizes: 2, 3 and 4, each of which has 2 filters. Every filter performs convolution on the sentence matrix and generates (variable-length) feature maps. Then 1-max pooling is performed over each map, i.e., the largest number from each feature map is recorded. Thus a univariate feature vector is generated from all six maps, and these 6 features are concatenated to form a feature vector for the penultimate layer. The final softmax layer then receives this feature vector as input and uses it to classify the sentence; here we assume binary classification and hence depict two possible output states. Source: Zhang, Y., & Wallace, B. (2015). A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification.

3.3.3 Pooling Layer

The second important building block of CNNs is a pooling layer. This layer is used to make the outputs less sensitive to the local variation in the inputs. This invariance to small local translation can decrease the spatial resolution and lead to underfitting in some applications. When accurate spatial features are not required, pooling can improve the performance of CNNs in extracting the features of interest. Further, pooling can reduce over-fitting since it decreases the number of dimensions and parameters [46]. In a sense, pooling takes subsamples from the outputs [45]. Similar to convolutional layers, pooling layers use a kernel (a rectangular receptive field) to apply an aggregation function such as maximum, average, L2-norm, or weighted average to summarize the values of the neurons within the pooling kernel. To have a pooling layer in CNNs, we need to determine the size of pooling kernels, the step of shifting, and the number of padding. Figure 3.6 depicts max pooling over a 3 × 3 matrix where the size of pooling kernel is 2 × 2 and the kernel shifts one pixel over the matrix (i.e., stride of 1).

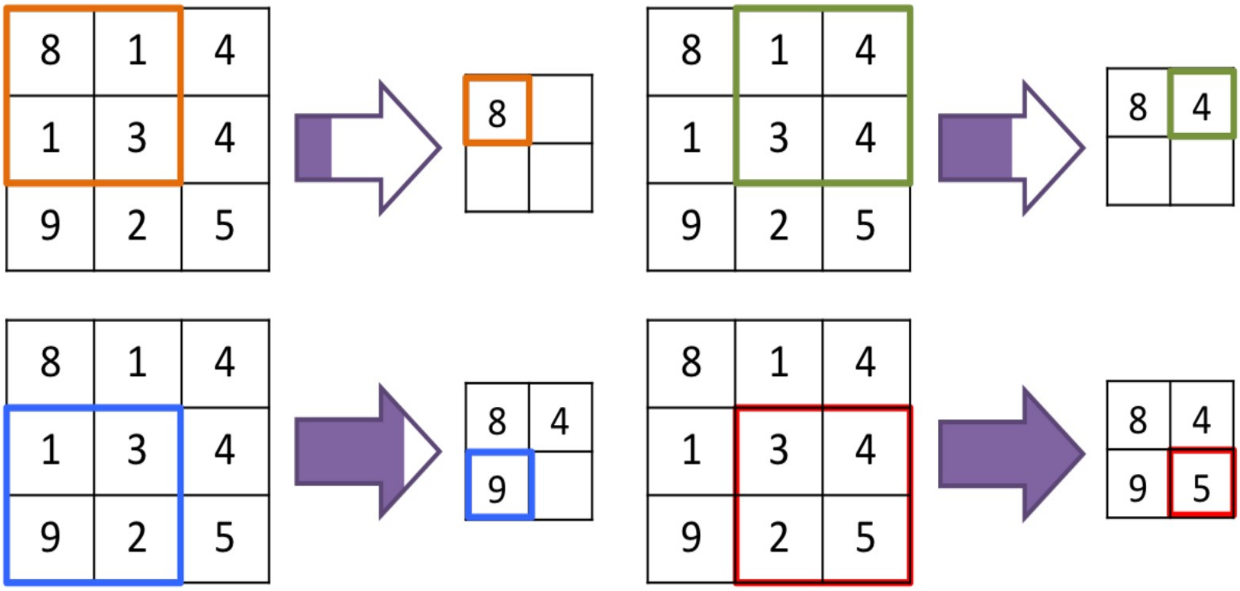


Figure 3.6: Pooling of a 3 × 3 image using a 2 × 2 kernel with a stride of 1; the maximum value of each window is subsampled.

3.3.4 Fully Connected Layer

A typical CNN consists of several convolutional layers where each convolutional layer is followed by a pooling layer. The last building block of CNNs is the fully connected layer, which is basically a traditional MLP. This component is used to either make a more abstract representation of the inputs by further processing of the features or classify the inputs based on the features extracted by preceding layers [53].

3.3.5 Activation Functions

Activation functions are important for an Artificial Neural Network to learn and understand the complex patterns. The main function of it is to introduce non-linear properties into the network. What it does is, it calculates the 'weighted sum' and adds direction and decides whether to 'fire' a particular neuron or not. There are several kinds of non-linear activation functions, like *Sigmoid*, *Tanh*, *ReLU* and *leaky ReLU*. The non linear activation function will help the model to understand the complexity and give accurate results.

3.3.6 CNN Framework

As discussed earlier, we have applied Convolutional Neural Networks algorithm on audio data. As the data is of one-dimensional, we cannot use the conventional CNN architecture used for image data. As a result we used 1-D convolutional layers instead of the most popular 2-D convolutional layers. All other layers like max-pooling and dense layers are used as it is. The convolutional neural network (CNN) architecture that has been implemented in the current study constitutes two convolutional layers and two fully connected layer, also known as dense layers. Among the two dense layers, the first one has 128 hidden neurons and the second one has 256 hidden neurons. For the current study, we tried to classify each audio file to a particular emotion class among 5 emotion classes and also to classify the gender of the voice. Thus we have 10 (5 emotions X 2 genders) output classes. As a result we added 10 *softmax* units in our last (output) layer to estimate the probability distribution of the classes. In our architecture, every convolutional layer is followed by a max-pooling layer. Each of the first and second convolutional layer is followed by a 1-D max-pooling layer of max-pooling window size of 7 and 4 respectively. The number of kernels (filters) is set to 64 and 128 for the first and second convolutional layer respectively. The sizes of the kernels that have been applied to the first and second convolutional layers are 5 and 3 respectively. Batch size of 16 is applied throughout the training process. Rectified Linear Units (ReLU) were used in convolutional layers and fully connected layers, except in the last dense layer, as activation functions to introduce non-linearity to the model. As the problem is a classification problem, we have used the Cross-entropy loss. Adam optimizer is also employed to minimize the loss function across the training data. The number of epochs is set to 100. The training procedure for this study was performed entirely on a CPU-based system, no GPU has been used for conducting any part of the training process.

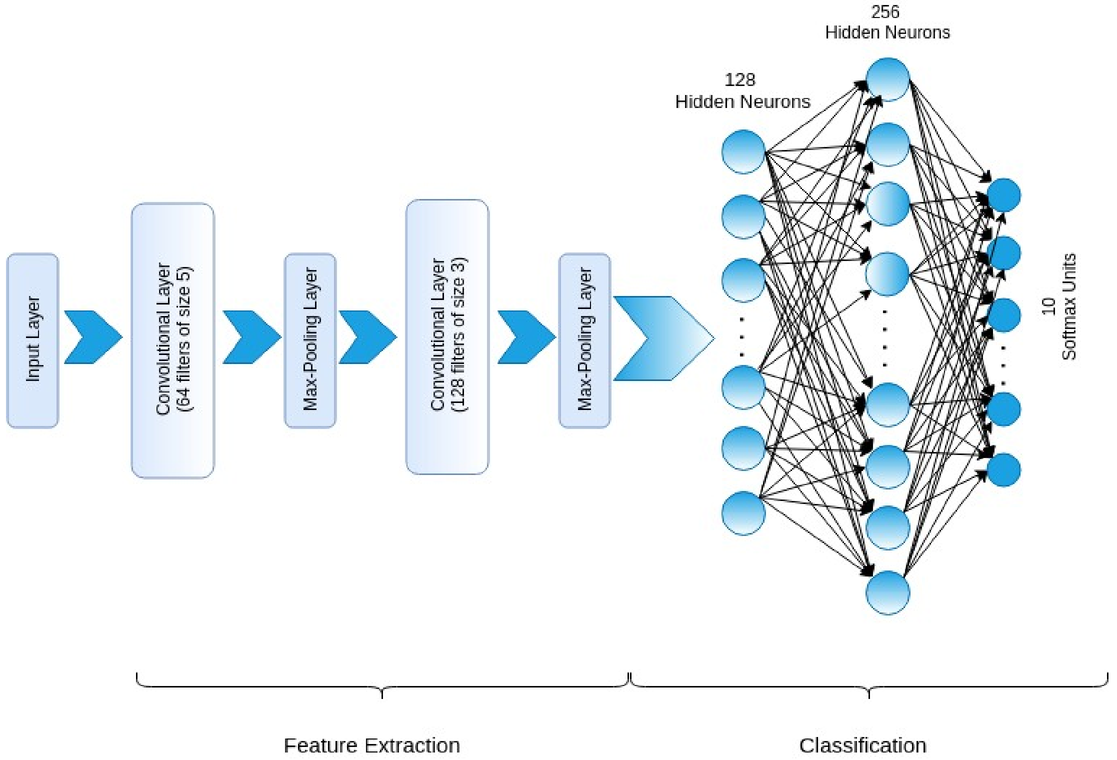


Figure 3.7: The baseline architecture of the CNN used in the current study to classify speech utterances based on their emotional states.

We have also used Dropout and Flatten function. Flatten function is used whenever we needed to reduce the dimension of the data which was output by a layer in the network and Dropout layer is used to reduce overfitting during the training process.

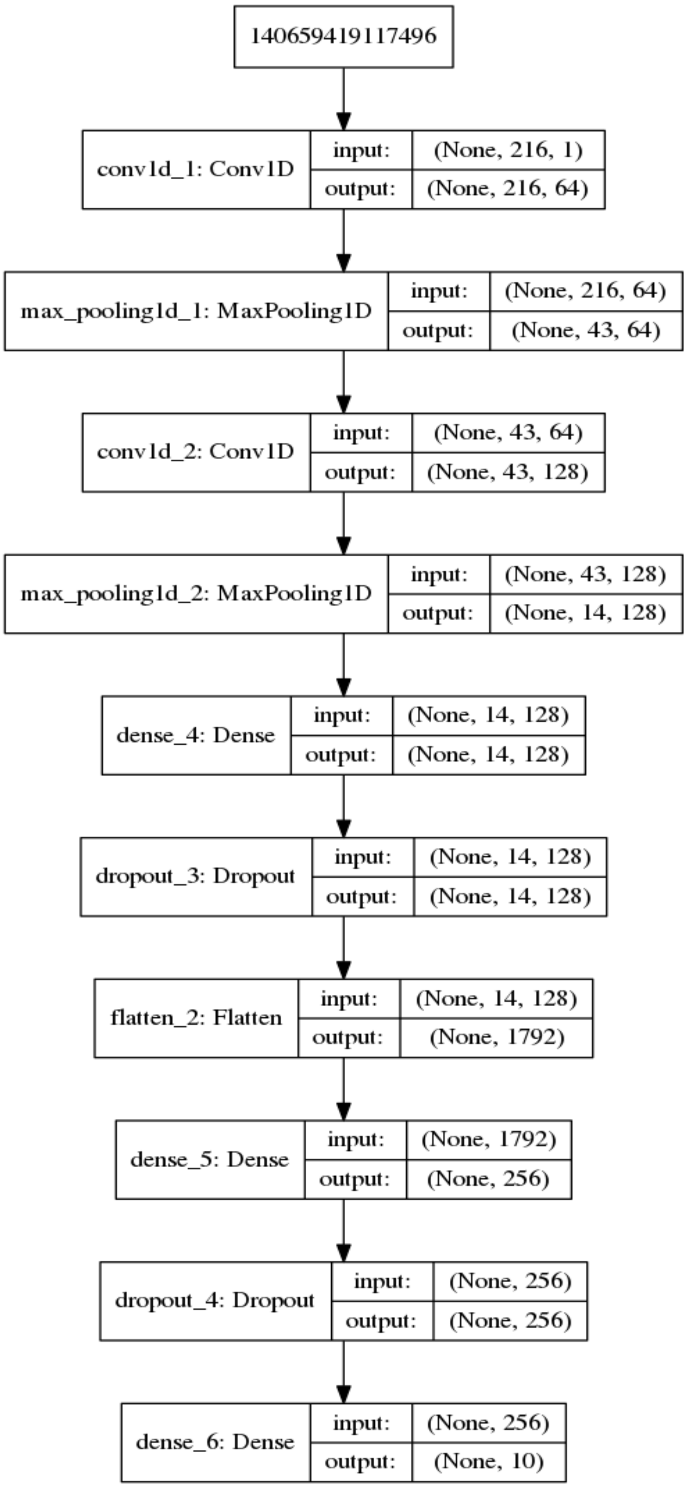


Figure 3.8: Detailed architecture which has been employed in this study

Dropout layers reduces over-fitting by dropping out or ignoring some of the neurons. We have used two dropout layers in our network architecture with each of them residing right after each of the two Dense layers. A dropout rate of 20% has been used in both of the two cases. Figure 3.8 gives a detailed overview of the network with the input and output dimensions of data in each of the layer in the network.

3.4 Experiments & Results

This section discusses the experiments performed in this study to classify emotion class and gender of the input audio data using the DNN and CNN systems (models) described in the above sections. To have a comparative discussion we restricted ourselves to 100 epochs for both the models.

3.4.1 Dataset Preparation

The dataset used to train both of these networks already has been discussed in the section 3.1. We encourage the readers to consult that section to have a detailed idea about the datasets. We have merged the audio files from the two datasets, SAVEE and RAVDESS, to produce the raw data. There are approximately 1900 audio files after merging. But we were not able to use all the audio files to train our networks as the emotion classes of the two datasets were not identical.

- The emotion classes reported in SAVEE database are Anger, Disgust, Fear, Happiness, Sadness, Surprise and Neutral.
- The emotion classes reported in RAVDESS database are Neutral, Calm, Happy, Sad, Angry, Fearful, Disgust, Surprised.

As we know Neutral emotion does not specifically portray any emotion specific feature, we discarded all the sentences belong to Neutral class from our raw dataset. Furthermore we considered only 5 main class of emotions namely, Calm, Happiness, Sadness, Fear and Anger. As a result we have approximately 1200 sentences in our raw database.

3.4.2 Train and Test Data Set

To train and test our models we need to split our raw dataset, which is described in the previous section, to form the training and test data. We have taken approximately 80% of the raw dataset as training data and the remaining 20% as test data to evaluate our models.

The performances of the neural network models on this training and test set have been demonstrated in the following sections.

3.4.3 Results of DNN Model

At first, we will look into the performance of the Deep Feed-forward Neural Network. We have reported Training vs. Test Accuracy graph, Training vs. Test Loss graph and two confusion matrices, one for emotion classification and another for gender classification.

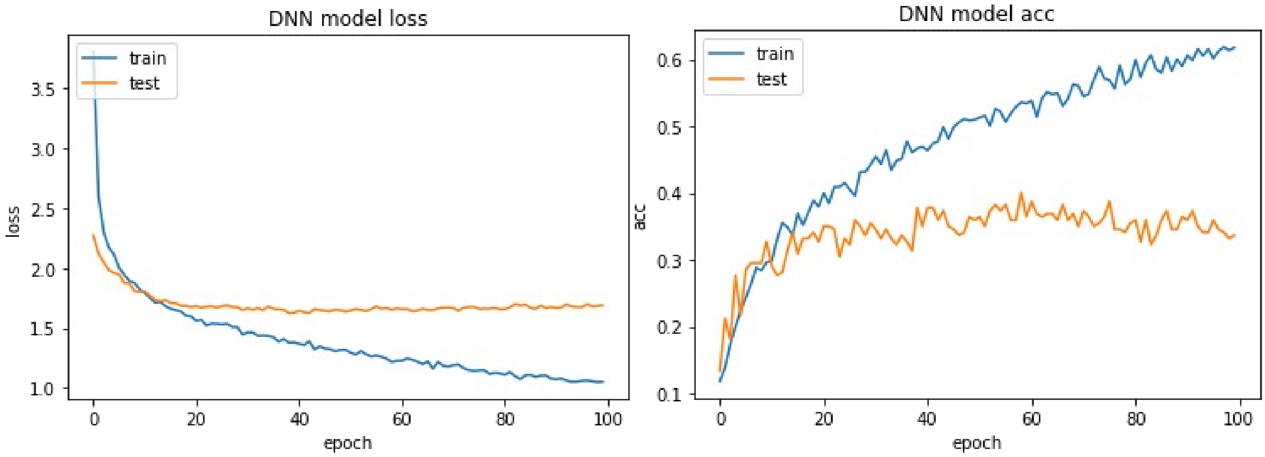


Figure 3.9: Training vs. Test Loss and Training vs. Test Accuracy graph for DNN model.

It is very much clear from the above graphs in Figure 3.9 that the model did not perform very well, in fact it is very clear that over-fitting happened in this case. Next we report the confusion matrix.

Figure 3.10 represents the overall confusion matrix for DNN model. If we analyse the confusion matrix, we can conclude that the overall performance of the DNN model is not good. We can see in the confusion matrix that *male_fearful*, *male_happy*, *male_sad* have been misclassified as *male_angry.* In addition to that considerable amount of *female_sad* and *male_sad* labels have been misclassified as *female_happy* and *male_fearful* respectively. Overall accuracy achieved by this model is *40.82%*.

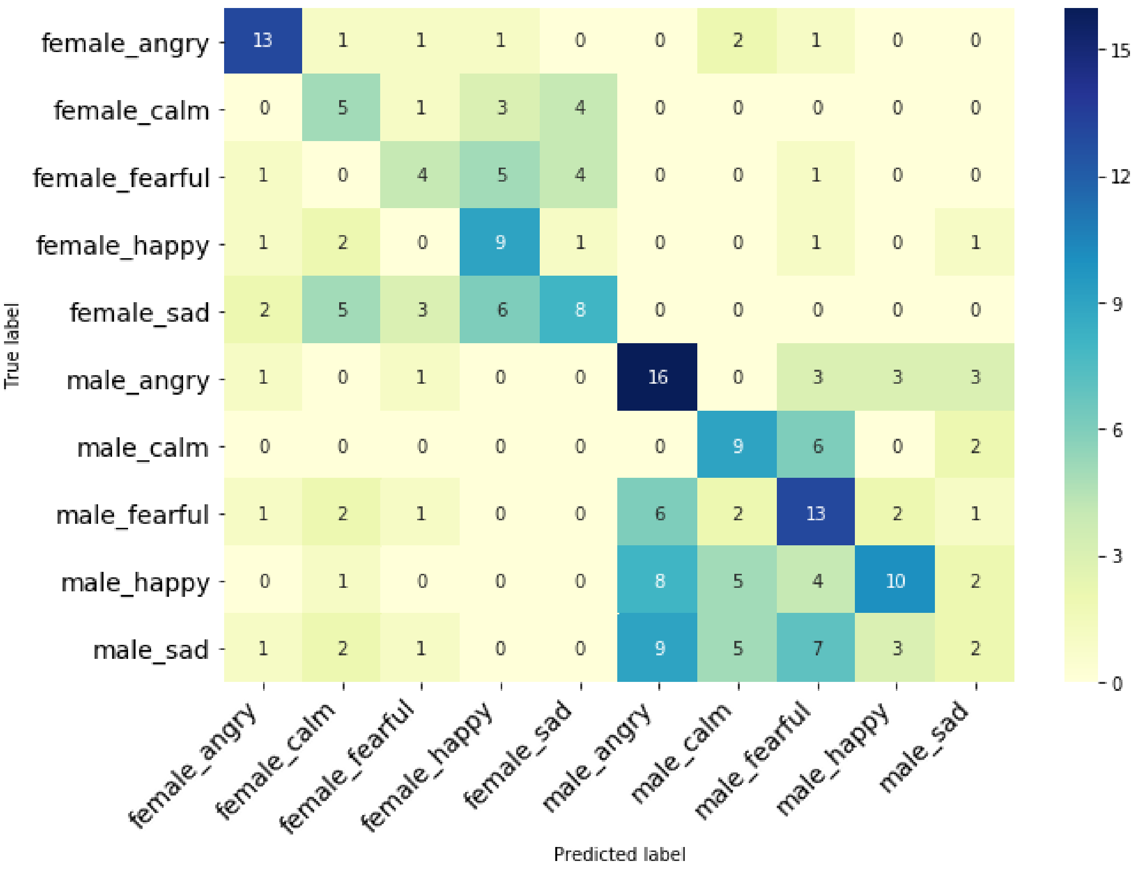


Figure 3.10: Confusion matrix for Emotion and Gender classification using DNN

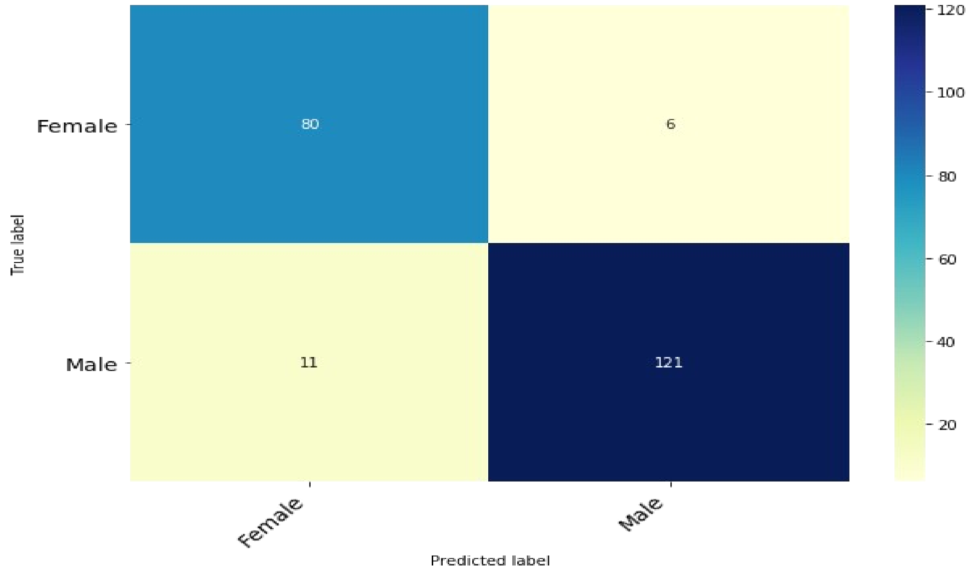


Figure 3.11: Confusion matrix for Gender classification using DNN model.

Figure 3.11 represents the confusion matrix only for gender labels, that is *male* and *female*. The classification performed by this model for gender labels is far better than overall classification.

3.4.4 Results of CNN Model

In this section we report the performance of the CNN model we implemented. We have Training vs. Test Accuracy graph, Training vs. Test Loss graph and two confusion matrices, one for emotion classification and another for gender classification as we have mentioned for the DNN model in the previous section.

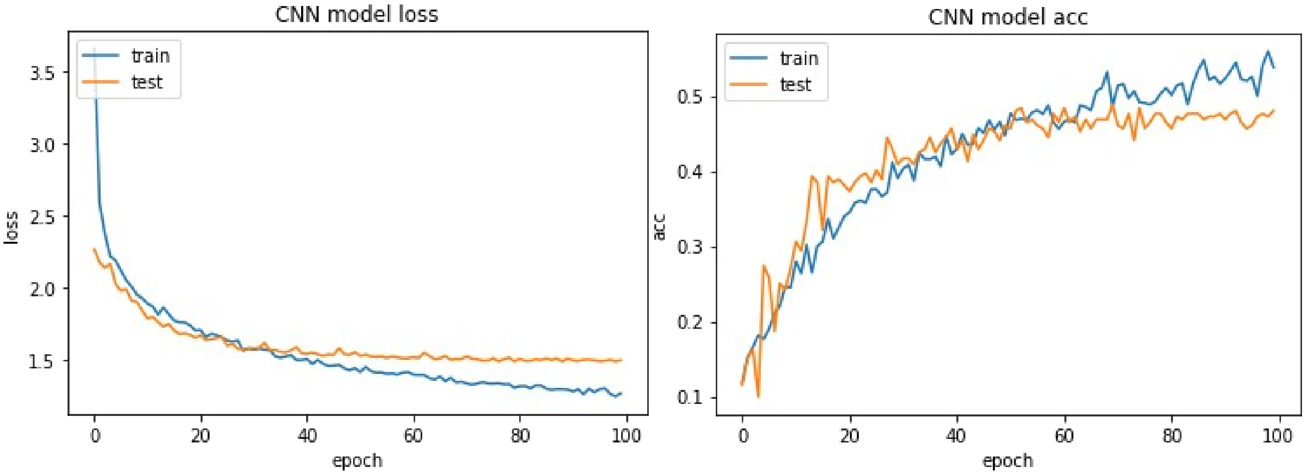


Figure 3.12: Training vs. Test Loss and Training vs. Test Accuracy graph for CNN model.

Unlike the DNN model, CNN model did not suffer from over-fitting as can be seen in Figure 3.12. Figure 3.13 and 3.14 represent the confusion matrices for overall classification and gender classification respectively. If we analyse the confusion matrix for the overall classification, it surely outperforms our DNN model as the misclassification rate is much lower in the case of CNN. Misclassification of *female_fearful* as *female_sad* is the only noticeable misclassification that happened in the whole confusion matrix. The accuracy for the overall classification is approximately *68.38%*. This accuracy was achieved by running the training algorithm for 2000 epochs.

The confusion matrix for gender classification has been reported in Figure 3.6. As we can see, the CNN model has classified most of the samples correctly.

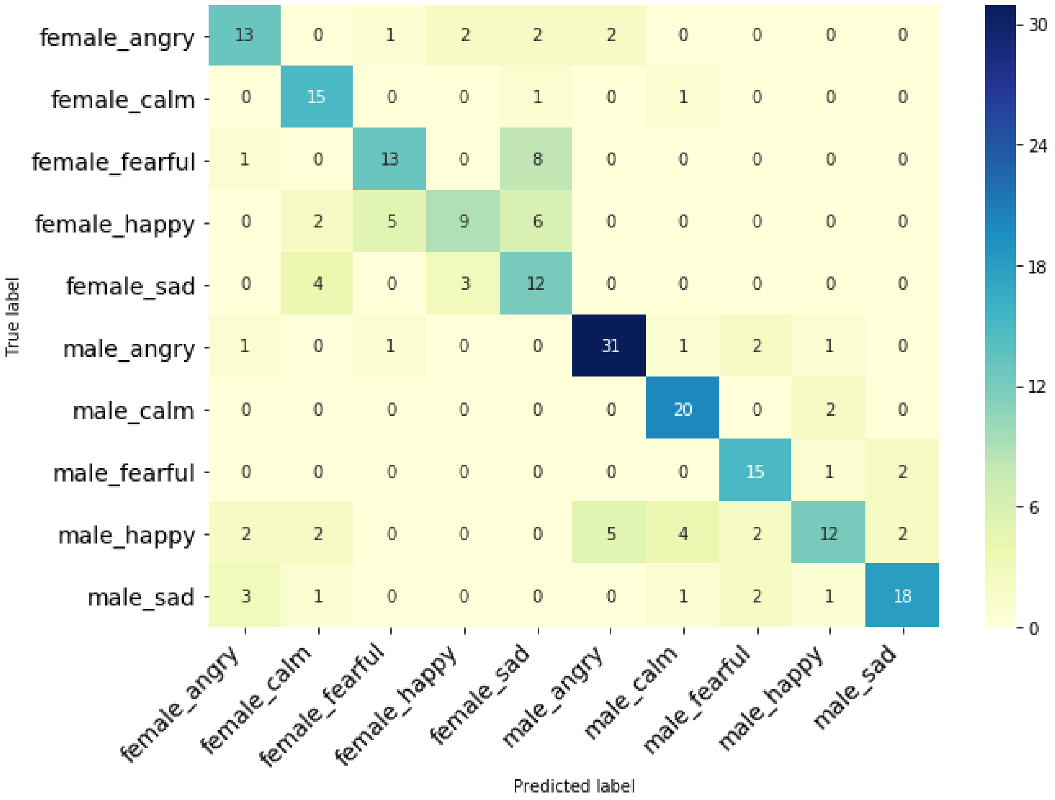


Figure 3.13: Confusion matrix for Emotion and Gender classification using CNN.

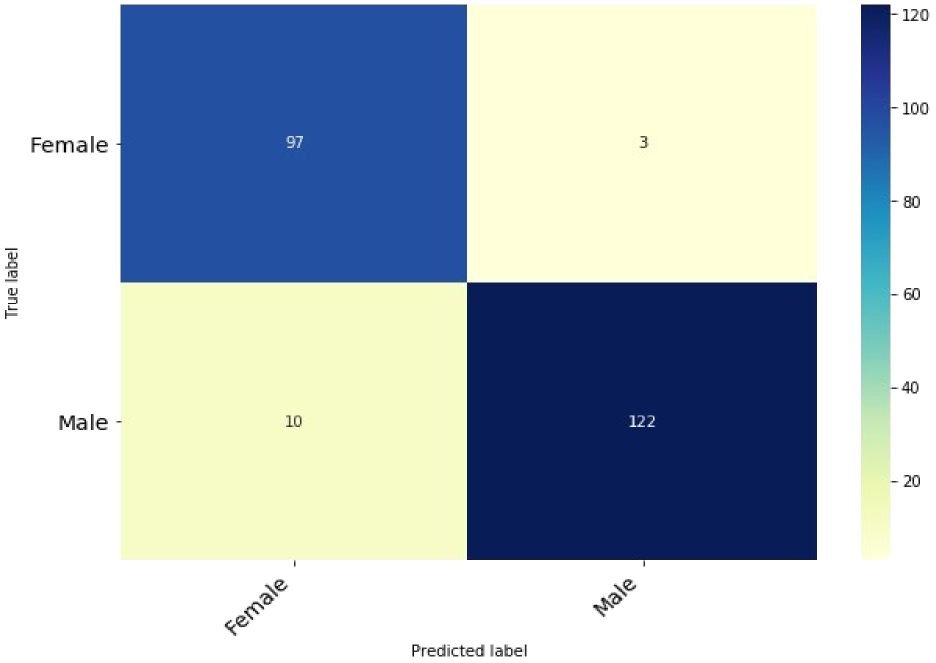


Figure 3.14: Confusion matrix for Gender classification using CNN model.

Chapter 4

Developing Speech Database of Emotion Words

4.1 Pre-processing of Audio Files

All audio, that is, our WAV files were resampled and filtered by an antialiasing FIR lowpass filter to have frequency rate of 44.1 kHz prior to any processing. Silences and non-voiced parts at the start and the end have been removed from the files.

4.2 Emotion Classification of Audios

The next step of the developing the database is to classify the emotion of each of the WAV files using the best classifier as discussed in the previous chapter.

4.3 Transcript Generation

To make our model robust, we made sure that we can build the database from the WAV files which does not have any transcript associated with it. We made use of IBM Speech to Text service to obtain the transcript of a given WAV file. The next few sections describe the workings of the IBM Speech to Text service.

4.3.1 Brief Overview of IBM Speech to Text Service

The IBM Speech to Text service provides Application Programming Interfaces (APIs) that use IBM's speech-recognition technologies to produce transcripts of spoken audio. Transcripts can be generated for almost all types of audio. All the well-known formats of speech files are supported. It also supports broadband and narrowband sampling rates. Along with the basic transcription text, it also provides other useful information such as confidence score and the start and end time of each of the words that constitute the audio file. The results that this system returns is of JSON format in the UTF-8 character set.

For speech recognition, the service supports synchronous and asynchronous HTTP Representational State Transfer (REST) interfaces. It also supports a WebSocket interface that provides a full-duplex, low-latency communication channel: Clients send requests and audio to the service and receive results over a single connection asynchronously.

4.3.2 Authentication

In any web-based services the most important and primary thing is Authentication. Authentication is the process of making sure if someone or something is, in fact, who or what it says itself to be. There are many types of authentication procedure available to secure web-based applications. IBM Cloud uses token-based Identity and Access Management (IAM) authentication procedure.

4.3.3 Recognition of Audio

In this step the user sends a request with an audio file and the format of the same over a WebSocket connection and it returns transcription results for recognition requests. Requests and responses travels over a single TCP connection that reduces complexity and offer efficient implementation, low latency, high throughput, and an asynchronous response.

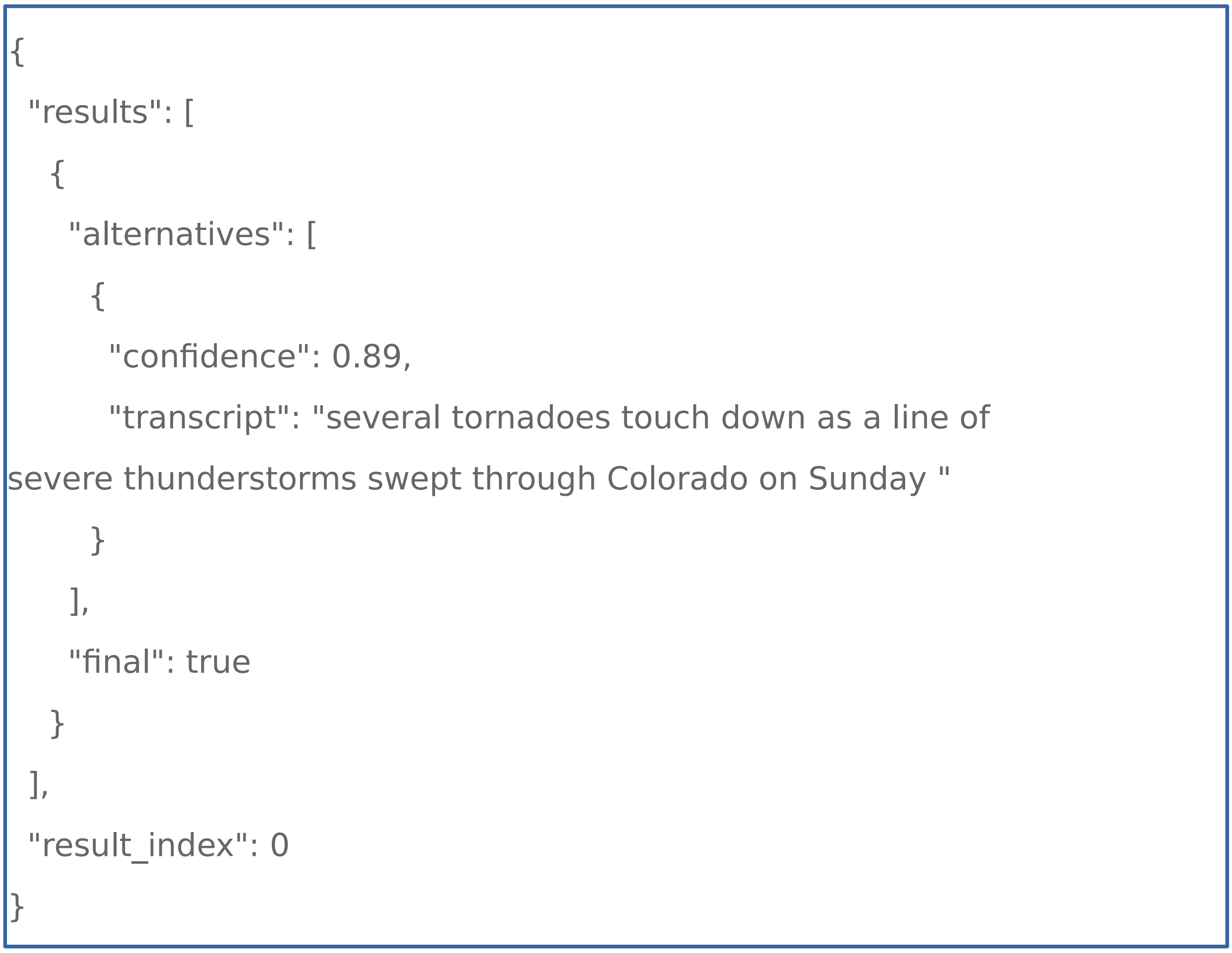
A maximum of 100 MB and a minimum of 100 bytes of audio per utterance can be passed per recognition request. Multiple utterances can also be sent over a single WebSocket connection. The service automatically detects the sequential order in which bytes of the incoming audio are arranged into larger numerical values when transmitted over digital links and for audio that includes multiple channels, it down-mixes the audio to one-channel mono during transcoding. By default, the service returns only final results for any request. To enable interim results, set the parameter to true.

4.3.4 Response

The IBM speech recognition APIs returns instances of *SpeechRecognitionResults* objects once it successfully recognizes the audio sent by the user. The information content of the responses depends on the parameters requested at the time of sending the audio recognition request. The results that this system returns is of JSON format in the UTF-8 character set irrespective of the interface.

To describe in more details on the response that we are getting from the service, let us consider the following example.

Suppose after submitting a request to the speech-to-text service, it returns the following response. The audio file contains a single sentence spoken by a male with no recognizable pauses between words.



```
{
  "results": [
    {
      "alternatives": [
        {
          "confidence": 0.89,
          "transcript": "several tornadoes touch down as a line of
severe thunderstorms swept through Colorado on Sunday "
        }
      ],
      "final": true
    }
  ],
  "result_index": 0
}
```

Figure 4.1: Sample response of a relatively simple request.

Now, let us analyse the response result. The result is actually a *SpeechRecognitionResults* object and for simple requests like the above one, it includes one *results* field and one *result_index* field.

- The *results* field contains an array of information about the transcription results. For this example, the *alternatives* field includes the transcript and the confidence field contains the confidence value with which the system generated the transcript. The *final* field has a value of *true* to indicate that these results will not change. This field contains the value false for interim results that are subject to change.
- The *result_index* field contains a unique identifier for the results. As the above displayed result is for a request with a single audio file with no pauses between words, and the request includes no additional parameters. So the service returns a single *result_index* field with a value of *0*, which is always the initial index.

If the interim results have been requested, the service returns multiple *results* fields. The indexes for interim results for the same audio always have the same value, and this value is ofcourse same as the index value of the final results for the same audio. If the sent audio contains recognizable pauses, the service can return multiple final results with different index values. In order to get the complete transcription of the audio, *transcript* field values of all the final results are concatenated ordering by the ascending value of *result_index* field.

If the input audio is more complex or the request includes additional parameters, the results can contain much more information. As shown in the following example, the response contains, in addition to all the above fields, the start and end time (in sec) of each recognizable words which constitutes the audio.

```
{
  "results": [
    {
      "keywords_result": {},
      "alternatives": [
        {
          "timestamps": [
            [
              "dogs",
              0.94,
              1.32
            ],
            [
              "are",
              1.32,
              1.41
            ],
            [
              "sitting",
              1.41,
              1.74
            ],
          ],
          "confidence": 0.95,
          "transcript": "dogs are sitting"
        }
      ],
      "final": true
    }
  ],
  "result_index": 0
}
```

Figure 4.2: Sample response of a more complex request.

4.4 POS Tagging and Word Segmentation from Audio

In this stage we tag the words based on their part of speech and segment the words from the audio using the transcript generated by the Text-to-Speech service. At first, Parts of Speech (POS) tagging of all the segmented words has been performed and we discarded the Proper Nouns (Names of places) as it conveys very little emotional features than adjectives or adverbs etc.

After this, we segment the words based on their start and end time in the audio files. We get the start and times of all the words from the results obtained from Text-to-Speech service described in the previous sections. we use this information to extract the words using *Pydub*, a Python library for audio processing.

4.5 Developing Database with Emotion-tagged Words

In this section, a flowchart has been introduced to develop the database. Figure 4.3 depicts the flow of work in order to produce the database.

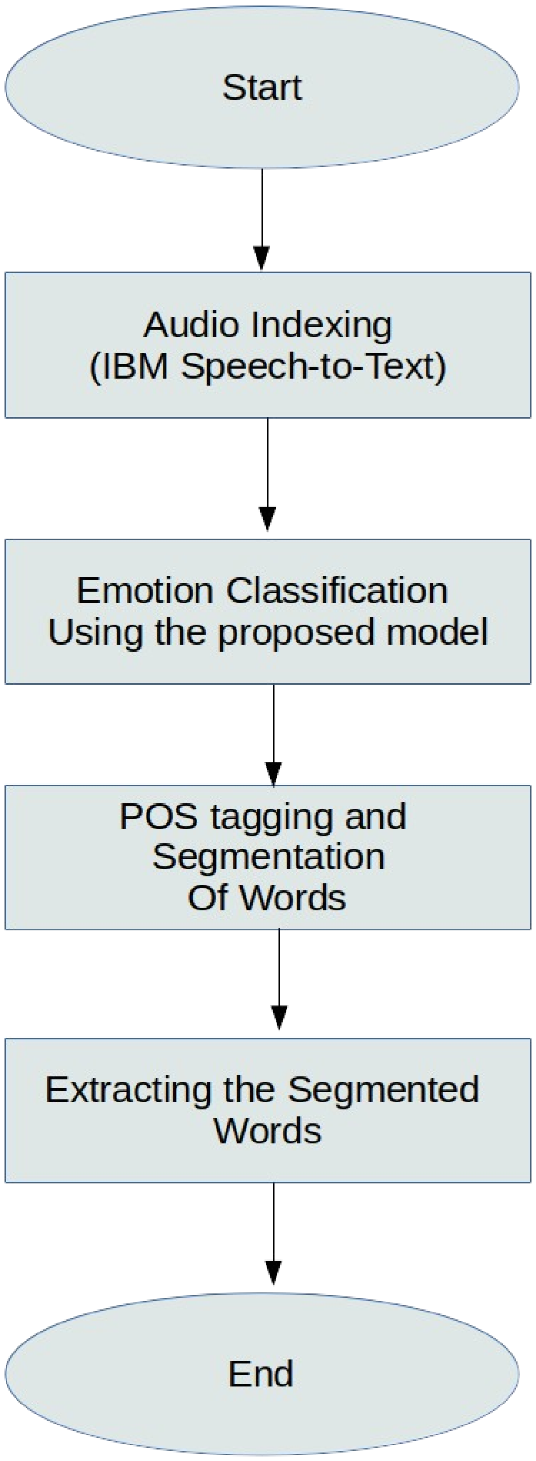
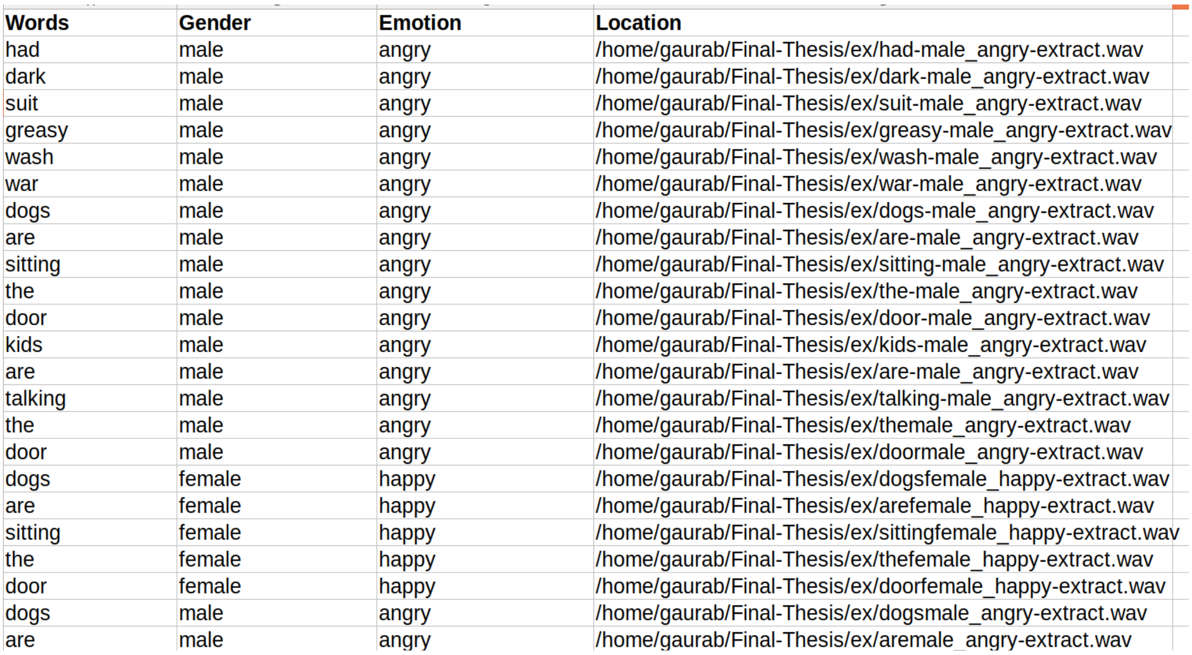


Figure 4.3: Flowchart of the proposed system to develop the database with emotion-tagged words.

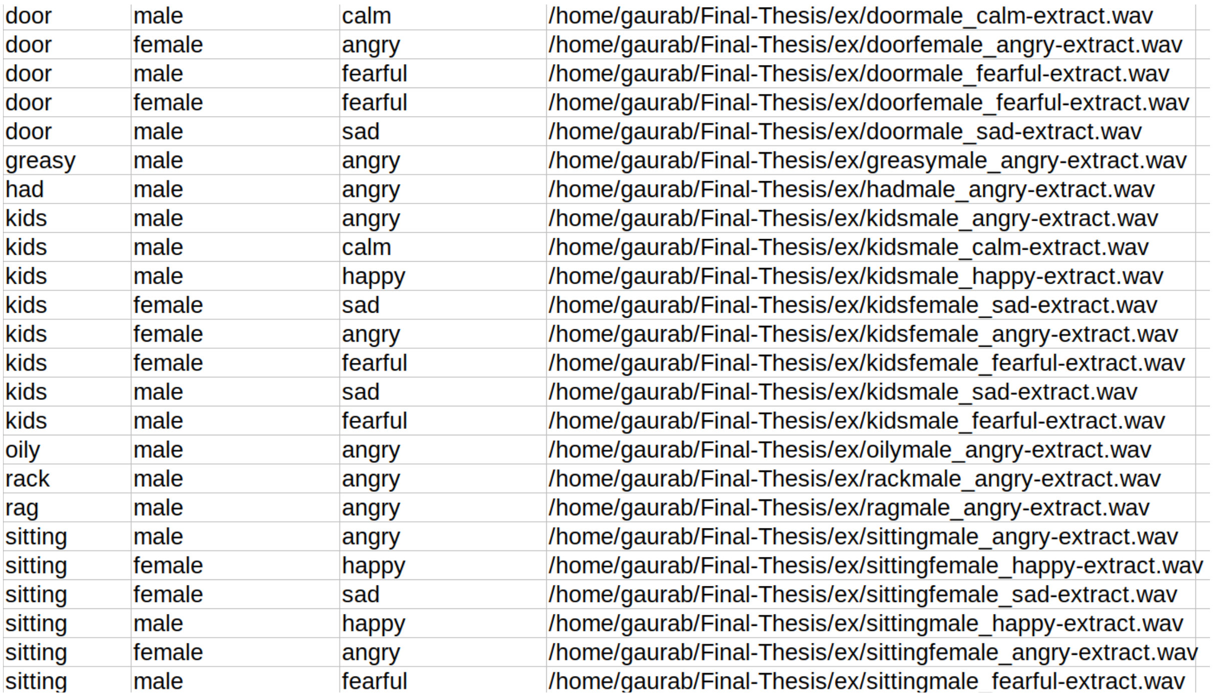


Words	Gender	Emotion	Location
had	male	angry	/home/gaurab/Final-Thesis/ex/had-male_angry-extract.wav
dark	male	angry	/home/gaurab/Final-Thesis/ex/dark-male_angry-extract.wav
suit	male	angry	/home/gaurab/Final-Thesis/ex/suit-male_angry-extract.wav
greasy	male	angry	/home/gaurab/Final-Thesis/ex/greasy-male_angry-extract.wav
wash	male	angry	/home/gaurab/Final-Thesis/ex/wash-male_angry-extract.wav
war	male	angry	/home/gaurab/Final-Thesis/ex/war-male_angry-extract.wav
dogs	male	angry	/home/gaurab/Final-Thesis/ex/dogs-male_angry-extract.wav
are	male	angry	/home/gaurab/Final-Thesis/ex/are-male_angry-extract.wav
sitting	male	angry	/home/gaurab/Final-Thesis/ex/sitting-male_angry-extract.wav
the	male	angry	/home/gaurab/Final-Thesis/ex/the-male_angry-extract.wav
door	male	angry	/home/gaurab/Final-Thesis/ex/door-male_angry-extract.wav
kids	male	angry	/home/gaurab/Final-Thesis/ex/kids-male_angry-extract.wav
are	male	angry	/home/gaurab/Final-Thesis/ex/are-male_angry-extract.wav
talking	male	angry	/home/gaurab/Final-Thesis/ex/talking-male_angry-extract.wav
the	male	angry	/home/gaurab/Final-Thesis/ex/themale_angry-extract.wav
door	male	angry	/home/gaurab/Final-Thesis/ex/doormale_angry-extract.wav
dogs	female	happy	/home/gaurab/Final-Thesis/ex/dogsfemale_happy-extract.wav
are	female	happy	/home/gaurab/Final-Thesis/ex/arefemale_happy-extract.wav
sitting	female	happy	/home/gaurab/Final-Thesis/ex/sittingfemale_happy-extract.wav
the	female	happy	/home/gaurab/Final-Thesis/ex/thefemale_happy-extract.wav
door	female	happy	/home/gaurab/Final-Thesis/ex/doorfemale_happy-extract.wav
dogs	male	angry	/home/gaurab/Final-Thesis/ex/dogsmale_angry-extract.wav
are	male	angry	/home/gaurab/Final-Thesis/ex/aremale_angry-extract.wav

Figure 4.4: Screenshot of the prepared database.

Figure 4.4 represents the database developed by our algorithm. The first column represents the words that have been spoken and second column represents the gender and third column represents the emotion in which the corresponding word has been spoken. The last column represents the location of the WAV file containing the utterance of the corresponding word in the specified emotion.

We have also group together same words spoken in different emotions as can be seen in the Figure 4.5. All the columns are same as previously mentioned.



door	male	calm	/home/gaurab/Final-Thesis/ex/doormale_calm-extract.wav
door	female	angry	/home/gaurab/Final-Thesis/ex/doorfemale_angry-extract.wav
door	male	fearful	/home/gaurab/Final-Thesis/ex/doormale_fearful-extract.wav
door	female	fearful	/home/gaurab/Final-Thesis/ex/doorfemale_fearful-extract.wav
door	male	sad	/home/gaurab/Final-Thesis/ex/doormale_sad-extract.wav
greasy	male	angry	/home/gaurab/Final-Thesis/ex/greasymale_angry-extract.wav
had	male	angry	/home/gaurab/Final-Thesis/ex/hadmale_angry-extract.wav
kids	male	angry	/home/gaurab/Final-Thesis/ex/kidsmale_angry-extract.wav
kids	male	calm	/home/gaurab/Final-Thesis/ex/kidsmale_calm-extract.wav
kids	male	happy	/home/gaurab/Final-Thesis/ex/kidsmale_happy-extract.wav
kids	female	sad	/home/gaurab/Final-Thesis/ex/kidsfemale_sad-extract.wav
kids	female	angry	/home/gaurab/Final-Thesis/ex/kidsfemale_angry-extract.wav
kids	female	fearful	/home/gaurab/Final-Thesis/ex/kidsfemale_fearful-extract.wav
kids	male	sad	/home/gaurab/Final-Thesis/ex/kidsmale_sad-extract.wav
kids	male	fearful	/home/gaurab/Final-Thesis/ex/kidsmale_fearful-extract.wav
oily	male	angry	/home/gaurab/Final-Thesis/ex/oilymale_angry-extract.wav
rack	male	angry	/home/gaurab/Final-Thesis/ex/rackmale_angry-extract.wav
rag	male	angry	/home/gaurab/Final-Thesis/ex/ragmale_angry-extract.wav
sitting	male	angry	/home/gaurab/Final-Thesis/ex/sittingmale_angry-extract.wav
sitting	female	happy	/home/gaurab/Final-Thesis/ex/sittingfemale_happy-extract.wav
sitting	female	sad	/home/gaurab/Final-Thesis/ex/sittingfemale_sad-extract.wav
sitting	male	happy	/home/gaurab/Final-Thesis/ex/sittingmale_happy-extract.wav
sitting	female	angry	/home/gaurab/Final-Thesis/ex/sittingfemale_angry-extract.wav
sitting	male	fearful	/home/gaurab/Final-Thesis/ex/sittingmale_fearful-extract.wav

Figure 4.5: Modified database where same words are grouped together across different emotional classes.

Chapter 5

Speech Synthesis From Text

5.1 Speech Synthesis Approaches

Speech synthesis, commonly known as Text To Speech, is the technique with which computers can speak. These methods of synthesizing speech have gone through a great evolution during the past two decades, and in this section some currently existing systems are introduced with their most used techniques. First, we review the most used in commercial environments which are the so called Unit Selection systems. Secondly, the Statistical Parametric Speech Synthesis is discussed, which has leveraged the speech synthesis research during the last decade. The section then concludes presenting the state of the art techniques of Deep Learning applied to speech synthesis.

5.1.1 Unit Selection Speech Synthesis

This type of synthesis has been in use for many years because it offers the best naturalness level, as it is based on real recorded speech (Hunt and Black, 1996). The way in which this system works is by concatenating segments of speech, which are usually called diphone. A diphone is a voice unit of the same size as a phoneme, defined in between of two phonemes (i.e. from the middle of a phoneme to the middle of the next one). Figure 5.1 exemplifies some hypothetic diphone boundaries compared to those of phonemes. The reason to do the division at the middle point of the phoneme is because it is the more stable point and the one least influenced by the co-articulation, which is the influence of neighboring phonemes to the current one.

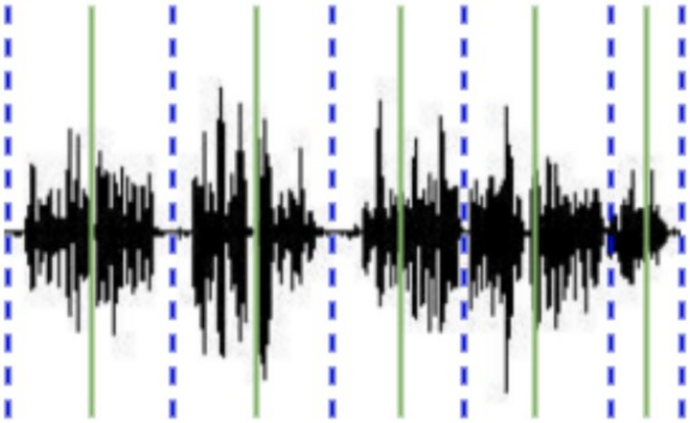


Figure 5.1: Voice stream where blue dashed lines show hypothetic phoneme divisions and green lines show hypothetic diphoneme divisions.

During concatenation to make the speech reconstruction the following issues need to be considered: discontinuities between the speech segments in phase, pitch, etc. and differences in prosody, which conveys the variation in duration or pitch (thus expressiveness) of the recorded segments with respect to the targeted prosody that should be achieved. To cope with these there are two approximations:

- Process the signal to smooth the discontinuities and force the prosody to match.
- Use a large database with many repetitions of the contained diphonemes, such that there is more variability to adapt to more possible con- texts and in reconstruction the chosen one is that matching better the prosodic requirements.

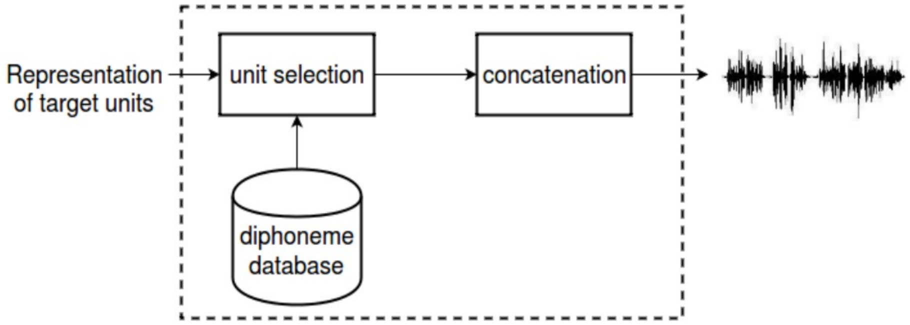


Figure 5.2: Unit selection scheme.

5.1.2 Statistical Parametric Speech Synthesis(SPSS)

The aim of this section is to make a brief review about the Statistical Parametric Speech Synthesis (SPSS). There are good works reviewing and gathering information about the SPSS systems in Zen, Tokuda, and Black (2009) and Ling et al. (2015). The following description is based on Ling et al. (2015) work. In this Text To Speech (TTS) approach, a set of stochastic generative acoustic models try to show a connection between text derived features and acoustic frames. In order to do so the speech signal is represented in a parametrised way, meaning that it gets encoded in a vocoder stage that creates an acoustic feature vector in a windowed fashion shifted every 5ms. The acoustic parameters of each phoneme in a given phonetic and prosodic context are represented by a stochastic generative model. Concretely context-dependent phoneme Hidden Markov Models (HMM) with single Gaussian state-output Probability Density Functions (PDF) are used. The phoneme contexts are defined using a decision-tree that clusters similar HMM output PDFs attending to the phonetic and prosodic features.

An HMM is a statistical Markov model with hidden (unobserved) states, which means that the state of every Markov state is not directly seen by the observer, but the outputs generated by this state (generated by means of the state-output PDF aforementioned) can be seen. The state transitions work the same way as in a Markov chain, with state-transition probabilities, which makes them suitable to model sequences.

Figure 5.3 is a schematic of the SPSS framework, where we can separate two stages: training and synthesis. During training, acoustic features of speech are extracted from the speech waveforms contained in a training data set (i.e. vocal tract and vocal source parameters). Context features are also extracted from the text transcriptions to build what are called the labels. Once we have the features, the context-dependent HMMs (λ^*) are estimated based on the Maximum Likelihood criterion:

$$\lambda^* = \underset{\lambda}{\operatorname{argmax}}\, p(\mathbf{y}|x, \lambda)$$

where $p(.)$ is a continuous PDF, $\mathbf{y}=\{y_1,y_2,\ldots,y_T\}$ is a sequence of acoustic features with T frames, being y_t the acoustic frame at time t. Finally, x = $\{x_1,\ldots,x_N\}$ is a sequence of linguistic context features, with N the number of phonemes. The acoustic feature vector is normally composed of static components and their first and second derivatives, such that:

$$\mathrm{y}=\{\mathrm{y}_{st}, \Delta\mathrm{y}_{st}, \Delta^2\mathrm{y}_{st}\} \tag{5.1}$$

Where Δ and Δ^2 stand for first and second derivatives. The complete acoustic feature set at time t can then be considered as a linear transform over the static feature sequence $\mathbf{y}_s = \{\mathbf{y}_{s1}, \mathbf{y}_{s2}, \ldots, \mathbf{y}_{sT}\}$:

$$\mathrm{y} = \mathbf{M}_y \mathbf{y}_s$$

$\mathbf{M}_y$ is determined by the expression to compute first and second derivatives used in equation 5.1.

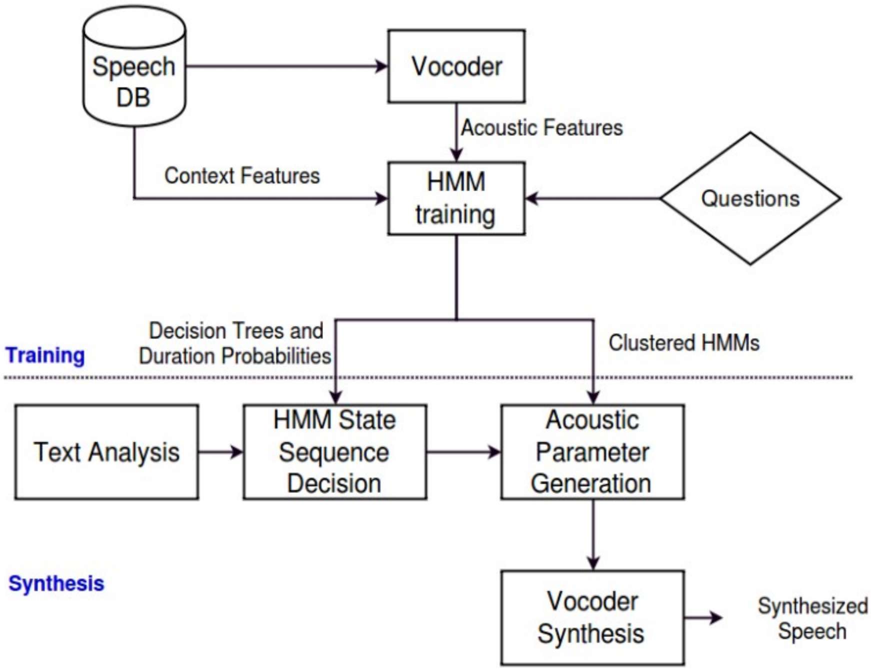


Figure 5.3: Block diagram of typical Statistical Parametric Speech Synthesis HMM-based systems. Based on Ling et al. (2015)[54].

In the synthesis part [55], at first, an arbitrarily given text which is to be synthesized is converted to a sequence of context-dependent labels, called *label* files, and then an utterance HMM is constructed by concatenating the context-dependent HMMs according to the label sequence. Second, state durations of the utterance HMM are determined based on the probability distribution functions of state duration. Third, the speech parameter generation algorithm which has been discussed as the Case 1 algorithm in [56], is used to generate the sequence of spectral and excitation parameters (Acoustic Parameters) that maximize their output probabilities. Finally, a speech waveform is synthesized directly from the generated spectral and excitation combinedly known as Acoustic Parameters, using the corresponding speech synthesis filter (Mel Log Spectrum Approximation (MLSA) filter [57] for Mel-cepstral coefficients), which is shown as Vocoder synthesis module in the above figure.

5.2 Model 3: Deep Learning in Speech Synthesis

Due to the flexibility in changing voice characteristics, like speaking style, speaker and adaptation capabilities, HMM-based TTS systems have gained enough popularity in the speech research community [58]. Nonetheless, there exists various drawbacks in HMM-based TTS [59]. Over the last few years Deep Neural Networks (DNNs) have been introduced to address some of the issues related to HMM-based TTS.

In SPSS system, the Acoustic Model which learns the mapping (relationship) between linguistic features and acoustic features, is a non-linear regression problem. The parametrisation process of HMMs is the most crucial part of the SPSS system and almost always it employs clustering of acoustically and linguistically-related context models using regression trees [60].

Deep feed-forward Neural Networks (DNN) has been extensively studied for mapping linguistic features to acoustic features directly without using regression trees as in the SPSS system [61,62,63]. The DNNs can be considered as a replacement for the regression tree (decision tree) used in HMM-based speech synthesis system as described in [64].

In the following section, a detailed implementation of DNN based speech synthesis system has been discussed.

5.2.1 Data Preparation

This section demonstrates the process of obtaining features from raw data which will be used to train and test our system. First, the textual features are explained. Their types and the process of conversion from raw text to *label* files is depicted. Then, acoustic features will be described, as well as the process they go through to produce the speech stream with the Vocoder.

5.2.1.1 Text to Label

At first, the raw text is processed into a representation that we call *label*, which is more convenient for our model to learn speech features. This representation is composed of a set of contextualized prosodic and phonetic features. Information about stressed syllables, position of the phoneme inside the current syllable, the position of the syllable in the word, etc. is embedded in these features. The features are a phonetic transcription of a few windowed phonemes, so that the synthesis of the current phoneme takes into account the surrounding phonemes for co-articulation purposes.

5.2.1.2 Acoustic Parameters

The proposed text-to-speech system in this work does not generate the voice waveform directly from the neural network itself, but it uses an intermediate speech generation module called Vocoder, shown in Figure 5.4.

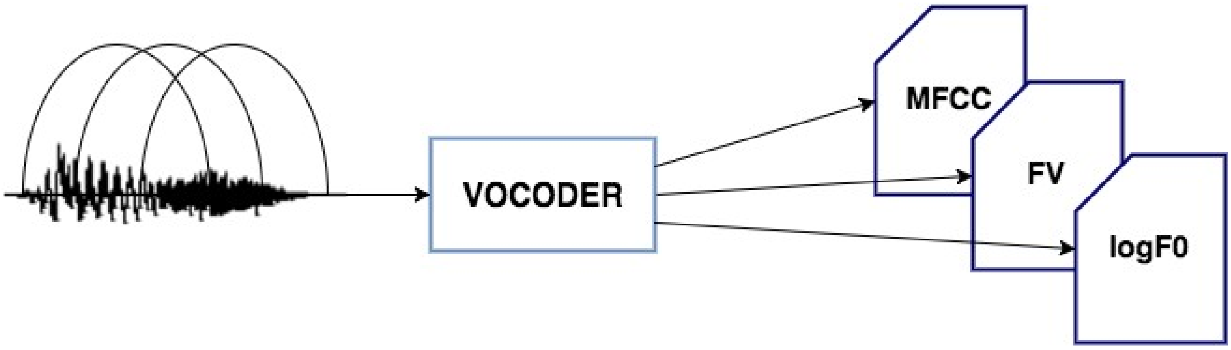


Figure 5.4: Schematic representation of the Vocoder for encoding the input voice with windowed frames into acoustic parameters.

The Vocoder takes the raw speech and extracts many acoustic frames composed of several features that describe the signal in a more convenient way. The extracted frames have good mathematical background. This is called Encoding.

On the other hand the Vocoder can also do the above explained process exactly in the other way around. Then this process would be called Decoding.
In this process, it takes the acoustic frames and converts them back to the speech signal. We have used WORLD vocoder [65] in this study.

5.2.2 Results and Discussions

The are two types of information required to produce a voice with good quality:

- The prosodic information: It considers intonation, phoneme duration, pauses between words, etc. characteristics that can make a huge effect on the voice naturalness.
- The acoustic information: Spectral estimation processed by the Vocoder system to generate the waveform. A good estimation is required for naturalness and also intelligibility.

The prosodic prediction is the first problem in the TTS design. At first, we tried to predict the phoneme duration. This will help us to know the amount of frames to be generated with the Vocoder, and then those frames will be generated out of the acoustic prediction system. So we have to train two separate models one for duration prediction and another for acoustic features (frames) predictions, where duration model has to be trained before acoustic frame prediction model :

1. Predict the duration for the current phoneme out of the encoded input linguistic features.
2. Predict the acoustic frame coefficients, for as many frames as dictated
by the duration prediction, also taking the linguistic features.

We have used simple DNN (Deep Feed-forward Neural Network) architecture to implement the above two models as our intention was to just implement a basic model and understand the backgrounds of Text-to-Speech system and not to carry out a comparative study across several architectures on the same.

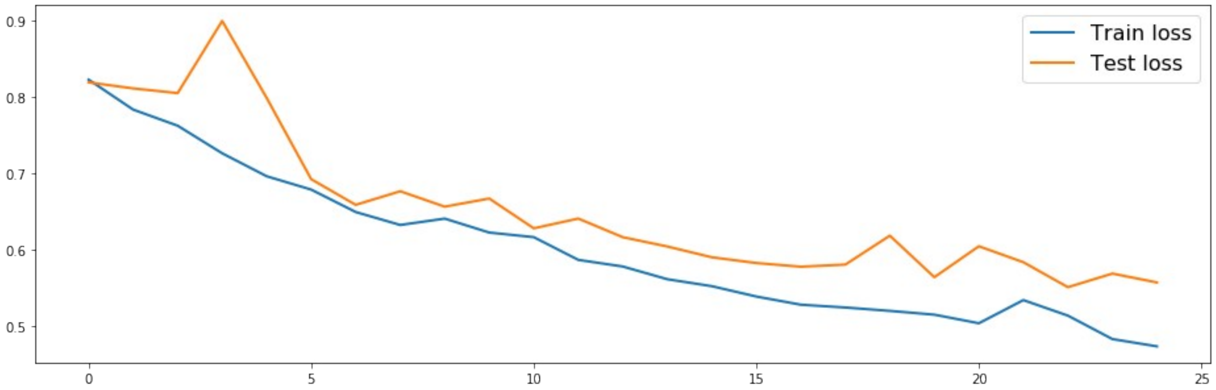


Figure 5.5: Train Vs. Test Loss graph for Duration Model.

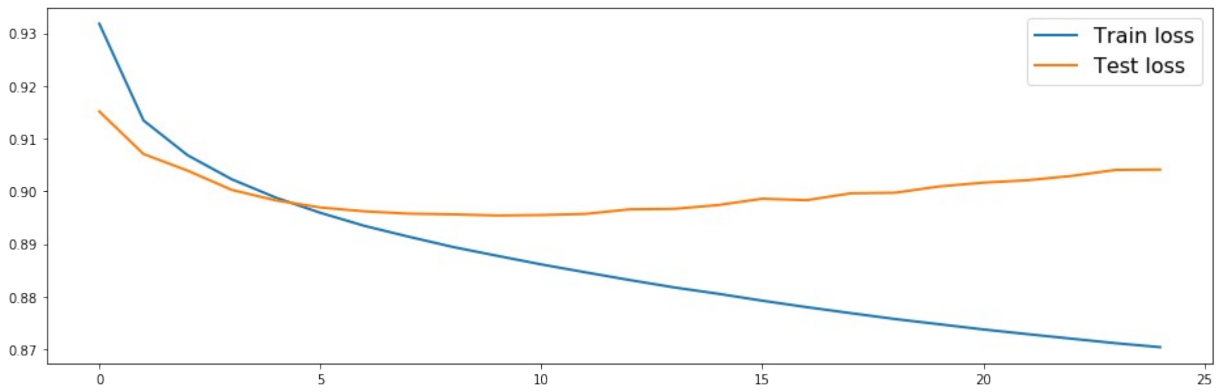


Figure 5.6: Train Vs. Test Loss graph for Acoustic Model.

Figure 5.5 and 5.6 report the graphs of train loss and test loss of both the Duration and Acoustic model respectively, that we have managed to obtain from our basic DNN model.

Chapter 6

Conclusion and Future Work

In this thesis titled “Extraction of Emotional Utterances by Employing Speech Emotion Recognition and Synthesis”, two classification models based on deep neural network, one using normal Deep Feed-forward Neural Network (DNN) and another using Convolutional Neural Network (CNN) architecture has been implemented and also a comparative study between these two models has been reported. Among the two models it has been shown that CNN model outperformed the DNN model. Also a database consisting of several emotion-tagged words has been developed as a part of this thesis work. And finally we implemented a basic DNN model for Text to Speech synthesis system.

The models have been developed from a training set which consisted only English language. It will be an interesting study to apply other languages to training the model and compare the performances for the same. It will also make the model more robust. In addition to that, we may implement other deep neural models like Recurrent Neural Network, specifically Long Short Term Memory (LSTM) in order to improve the classifier system.

As we have mentioned earlier that we have developed a database which contains emotion-tagged words but the size of the database is not very large. As the available datasets for speech emotion research are very less in number, we had a limited number of testing samples and hence it affected the development of the proposed database.

For the Text to Speech system, we reported a very basic system to get us familiar with the whole speech synthesis process, so there are ample space for improvement on that basic system, such as incorporating more naturalness in the synthesised voice.

References

[1] A modality is defined as channel or form for rendering a thought, concept, or action (Coutaz and Caelen, 1991).

[2] Stuart Jonathan Russell, Peter Norvig, John F Canny, Jitendra M Malik and Douglas D Edwards. *Artificial intelligence: a modern approach*, volume 2. Prentice hall Upper Saddle River, 2003.

[3] Demis Hassabis, Dharshan Kumaran, Christopher Summerfield, and Matthew Botvinick. Neuroscience-inspired artificial intelligence. *Neuron, 95(2)*: pages 245–258, 2017.

[4] Marcel Van Gerven. Computational foundations of natural intelligence. *Frontiers in Computational Neuroscience*, 11:112, 2017.

[5] Paul R Cohen and Edward A Feigenbaum. *The handbook of artificial intelligence*, volume 3. Butterworth-Heinemann, 2014.

[6] Ray Kurzweil. The singularity is near. *Gerald Duckworth & Co*, 2010.

[7] Marvin Minsky. The emotion machine: Commonsense thinking, artificial intelligence, and the future of the human mind. *Simon and Schuster*, 2007.

[8] M. El Ayadi, M. S. Kamel, and F. Karray, "Survey on speech emotion recognition: Features, classification schemes, and databases," *Pattern Recognition*, vol. 44, no. 3, pages. 572–587, Mar. 2011.

[9] Dong Yu and Li Deng., Automatic Speech Recognition. *Springer*,2016.

[10] Lawrence R Rabiner and Biing-Hwang Juang. *Fundamentals of speech recognition*, volume 14. PTR Prentice Hall Englewood Cliffs, 1993.

[11] Lalit R Bahl, Frederick Jelinek, and Robert L Mercer. A maximum likelihood approach to continuous speech recognition. In *Readings in speech recognition*, pages 308–319. Elsevier, 1990.

[12] Stephen E Levinson, Lawrence R Rabiner, and Man Mohan Sondhi. An introduction to the application of the theory of probabilistic functions of a markov process to automatic speech recognition. *The Bell System Technical Journal*, 62(4), pages 1035–1074, 1983.

[13] Su-Lin Wu, ED Kingsbury, Nelson Morgan, and Steven Greenberg. Incorporating information from syllable-length time scales into automatic speech recognition. In Acoustics, Speech and Signal Processing, 1998. *Proceedings of the 1998 IEEE International Conference* on, volume 2, pages 721–724. IEEE, 1998.

[14] Vaibhava Goel and William J Byrne. Minimum bayes-risk automatic speech recognition. *Computer Speech & Language*, 14(2): pages 115–135, 2000.

[15] Picard, R.W. Affective Computing. *M.I.T. Press*, Cambridge, MA. 1997.

[16] Picard R. W., Healey J., Affective Wearables, *Personal Technologies* Vol 1, No. 4, pages 231-240. 1997.

[17] Picard R.W., Affective Computing for HCI. In *Proc. of the 8th International Conference on Human- Computer Interaction: Ergonomics and User Interfaces*-Volume I. Lawrence Erlbaum Associates, Inc. 1999.

[18] Picard R.W., Vyzas E., Healey J. Toward Machine Emotional Intelligence -Analysis of Affective Physiological State. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol 23 No. 10, 2001.

[19] Norman, D.A. Emotional Design: Why we love (or hate) everyday things. *Basic Books*. 2003.

[20]. F. Dipl and T. Vogt, "Real-time Automatic Emotion Recognition from Speech", 2010.

[21]. S. Lugovic, I. Dunder, and M. Horvat, Techniques and applications of emotion recognition in speech, 2016 *39th Int. Conv. Inf. Commun. Technol.818181 Electron. Microelectron.* MIPRO 2016 - Proc., November 2017, pages 1278–1283, 2016.

[22]. B. Schuller, G. Rigoll, and M. Lang, "Speech emotion recognition combining acoustic features and linguistic information in a hybrid support vector machine - belief network architecture," *Acoust. Speech, Signal Process.*, vol. 1, pages 577–580, 2004.

[23]. R. W. Picard, "Affective Computing for HCI," In *HCI* (1), pages 829–833, 1999.

[24]. F. Ren, "From cloud computing to language engineering, affective computing and advanced intelligence," *International Journal of Advanced Intelligence*, vol. 2(1), pages 1–14, 2010.

[25]. Changqin Quan, Fuji Ren, "An Exploration of Features for Recognizing Word Emotion", Proceedings of the *23rd International Conference on Computational Linguistics (Coling 2010)*, pages 922–930, Beijing, August 2010

[26] F. Burkhardt, A. Paeschke, M. Rolfes, W. Sendlmeier, B. Weiss, A database of German emotional speech, in: *Proceedings of the Interspeech 2005*, Lisbon, Portugal, pages 1517–1520, 2005.

[27] R.Banse, K.Scherer, Acoustic profiles in vocal emotion expression, *J. Pers. Soc. Psychol.* 70 (3), pages 614–636, 1996.

[28] L. Chen, X. Mao, Y. Xue, and L. L. Cheng, “Speech emotion recognition: Features and classification models,”Digit. *Signal Process.*, vol. 22, no. 6, pages 1154–1160, Dec. 2012.

[29] H. Cao, R. Verma, and A. Nenkova, “Speaker-sensitive emotion recognition via ranking: Studies on acted and spontaneous speech,” *Comput. Speech Lang.*, vol. 28, no. 1, pages 186–202, Jan. 2015.

[30] L. Chen, X. Mao, Y. Xue, and L. L. Cheng, “Speech emotion recognition: Features and classification models”, *Digit. Signal Process.*, vol. 22, no. 6, pages 1154–1160, Dec. 2012.

[31] T. L. Nwe, S. W. Foo, and L. C. De Silva, “Speech emotion recognition using hidden Markov models,” *Speech Communication*, vol. 41, no. 4, pages 603–623, Nov. 2003.

[32] S. Wu, T. H. Falk, and W.-Y. Chan, “Automatic speech emotion recognition using modulation spectral features,” *Speech Communication*, vol. 53, no. 5, pages 768–785, May 2011.

[33] J. Rong, G. Li, and Y.-P. P. Chen, “Acoustic feature selection for automatic emotion recognition from speech,” *Information Processing and Management*, vol. 45, no. 3, pages 315–328, May 2009.

[34] C.-H. Wu and W.-B. Liang, “Emotion Recognition of Affective Speech Based on Multiple Classifiers Using Acoustic-Prosodic Information and Semantic Labels,” *IEEE Trans. Affective Computing*, vol. 2, no. 1, pages 10–21,Jan. 2011.

[35] S. S. Narayanan, “Toward detecting emotions in spoken dialogs,” *IEEE Trans. Speech Audio Process.*, vol. 13,no. 2, pages 293–303, Mar. 2005.

[36] B. Yang and M. Lugger, “Emotion recognition from speech signals using new harmony features,” *Signal Processing*, vol. 90, no. 5, pages 1415–1423, May 2010.

[37] E. M. Albornoz, D. H. Milone, and H. L. Rufiner, “Spoken emotion recognition using hierarchical classifiers”, *Computer Speech and Language*, vol. 25, no. 3, pages 556–570, Jul. 2011.

[38] J.-H. Yeh, T.-L. Pao, C.-Y. Lin, Y.-W. Tsai, and Y.-T. Chen, “Segment-based emotion recognition from continuous Mandarin Chinese speech,” *Computers in Human Behaviour*, vol. 27, no. 5, pages 1545–1552, Sep. 2011.

[39] M. Grimm, K. Kroschel, E. Mower, and S. Narayanan, “Primitives-based evaluation and estimation of emotions in speech,” *Speech Commun.*, vol. 49, no. 10–11, pages 787–800, Oct. 2007.

[40] J. P. Arias, C. Busso, and N. B. Yoma, “Shape-based modeling of the fundamental frequency contour for emotion detection in speech,” *Comput. Speech Lang.*, vol. 28, no. 1, pages 278–294, Jan. 2014.

[41] Sanaul Haq, Philip JB Jackson, and J Edge. Speaker-dependent audio-visual emotion recognition. In *AVSP*, pages 53–58, 2009.

[42] Ekman, P., “Universals and cultural differences in facial expressions of emotion”, *Nebraska Symposium on Motivation,* pages 207-283, 1972.

[43] Zeng, Z., Pantic, M., Roisman, G.I. and Huang, T.S., “Survey of Affect Recognition Methods: Audio, Visual, and Spontaneous Expressions”, *IEEE Trans. PAMI*, 31(1), pages 39-58, 2009.

[44] Livingstone SR, Russo FA (2018) The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. *PLoS* ONE 13(5): e0196391.

[45] Aurélien Géron. *Hands on machine learning with scikit-learn and tensorflow*, 2017.

[46] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep learning. *MIT press*, 2016.

[47] Tom M Mitchell et al. Machine learning. *McGraw-Hill International Edition*, 1997.

[48] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[49] Steve Lawrence, C Lee Giles, Ah Chung Tsoi, and Andrew D Back. Face recognition: A convolutional neural-network approach. *IEEE transactions on neural networks*, 8(1):98–113, 1997.

[50] Yann LeCun, Bernhard E Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne E Hubbard, and Lawrence D Jackel. Handwritten digit recognition with a back-propagation network. In *Advances in neural information processing systems*, pages 396–404, 1990.

[51] Wayne Xiong, Jasha Droppo, Xuedong Huang, Frank Seide, Mike Seltzer,
Andreas Stolcke, Dong Yu, and Geoffrey Zweig. The microsoft 2016 conversational speech recognition system. In *Acoustics, Speech and Signal Processing (ICASSP)*, 2017 IEEE International Conference on, pages 5255–5259. IEEE, 2017.

[52] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, pages 649–657, 2015.

[53] Yann LeCun, Yoshua Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995.

[54] Zhen-Hua Ling, Shi-Yin Kang, Heiga Zen, Andrew Senior, Mike Schuster, Xiao-Jun Qian, Helen Meng, and Li Deng. “Deep Learning for Acoustic Modeling in Parametric Speech Generation”, in *IEEE SIGNAL PROCESSING MAGAZINE*, pages 35-52, May 2015.

[55] Heiga Zen, Takashi Nose, Junichi Yamagishi, Shinji Sako, Takashi Masuko, Alan W. Black, Keiichi Tokuda, “The HMM-basedSpeech SynthesisSystem (HTS) Version 2.0”, in *6th ISCA Workshop on Speech Synthesis*, Bonn, Germany, August 22-24, 2007

[56] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, “Speech parameter generation algorithms for HMM-based speech synthesis,” in *Proc. ICASSP*, pages 1315–1318, 2000.

[57] S. Imai, “Cepstral analysissynthesis on the mel frequency scale,” in *Proc. ICASSP*, pages 93–96, 1983.

[58] Yamagishi, J., Kobayashi, T., Nakano, Y., Ogata, K., Isogai, J.: Analysis of speaker adaptation algorithms for HMM-based speech synthesis and a constrained SMAPLR adaptation algorithm. *Trans. Audio Speech Lang. Proc.* 17(1), pages 66–83, 2009.

[59] Zen, H., Senior, A., Schuster, M.: Statistical parametric speech synthesis using deep neural networks. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 7962–7966, 2013.

[60] Zhizheng Wu, Oliver Watts, Simon King, Merlin: An Open Source Neural Network Speech Synthesis System. In *9th ISCA Speech Synthesis Workshop* 13-15 Sep 2016, Sunnyvale, USA.

[61] H. Lu, S. King, and O. Watts, “Combining a vector space representation of linguistic context with a deep neural network for text-to-speech synthesis,” In Proc. t*he 8th ISCA Speech Synthesis Workshop (SSW)*, pages 281–285, 2013.

[62] Y. Qian, Y. Fan, W. Hu, and F. K. Soong, “On the training aspects of deep neural network (DNN) for parametric TTS synthesis,” in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2014, pages 3829–3833.

[63] K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda, “The effect of neural networks in statistical parametric speech synthesis,” in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2015, pages 4455–4459.

[64] O. Watts, G. E. Henter, T. Merritt, Z. Wu, and S. King, “From HMMs to DNNs: where do the improvements come from?” in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2016.

[65] M. MORISE, F. YOKOMORI, and K. OZAWA, “WORLD: a vocoder-based high-quality speech synthesis system for real-time applications,” *IEICE transactions on information and systems*, 2016.