Emotion Tagging In An Audio Signal Using Weakly Supervised learning

DISSERTATION

Submitted in partial fulfillment of the requirements of
M.Tech Data Science and Engineering Degree programme

By

Naga Srimouli Borusu
ID No. 2019AH04075

Under the supervision of:

Koti Reddy Mutyam Gurunadha
Senior Technical Manager
Netcracker Technology India Pvt Ltd.

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE
Pilani (Rajasthan) INDIA

February 15, 2022

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

CERTIFICATE

This is to certify that the Dissertation entitled, "*Emotion Tagging In An Audio Signal Using Weakly Supervised learning*" and submitted by Naga Srimouli Borusu ID No. 2019AH04075 in partial fulfillment of the requirements of DSECLZG628T Dissertation embodies the work done by him under my supervision.

Signature of the Supervisor
Place: Hyderabad
Date: 21 - January - 2022

Koti Reddy Mutyam Gurunadha
Senior Technical Manager,
Netcracker Technology India Pvt Ltd.

“The key to artificial intelligence has always been the representation.”

–Jeff Hawkins

BIRLA INSTITUTE OF TECHNOLOGY SCIENCE, PILANI

SECOND SEMESTER 2020-21

DSECLZG628T DISSERTATION

Dissertation Title	:	Emotion Tagging In An Audio Signal Using Weakly Supervised learning
Name of Supervisor	:	Koti Reddy Mutyam Gurunadha
Name of Student	:	Naga Srimouli Borusu
ID of the Student	:	2019AH04075

Abstract

Emotions are vital in human-to-human communication and connection because they allow individuals to express themselves in ways beyond the scope of language. The computer's ability to read human emotions is critical in a variety of applications. Compared to other machine/deep learning research domains, such as computer vision, audio analysis has received less attention. The majority of existing research in this area is focused on supervised learning algorithms, with limited emphasis on weakly-supervised approaches. Obtaining the tagged data sets that subject matter experts have annotated is an expensive and time-consuming process. To address this issue, weakly supervised approaches can use labels generated by non-experts to aid the model train with minimal input of annotated tags from subject matter experts. The project compares the efficiencies of different deep learning techniques to predict the emotion labels of an input audio signal.

Keywords:

Emotion tagging, Audio Analysis, Weakly Supervised learning, Emotion, Autoencoders, CNN.

Acknowledgements

First and foremost, I want to express my gratitude to Mr. Koti Reddy Mutyam Gurunadha, my M.Tech supervisor, for his helpful advice and unwavering support during my studies. I'd also want to thank Ms.Seetha Parameswaran for her support and technical insights, without which completing this work could've been a Herculean endeavor. Thank everyone on my team at Netcracker Technology India Pvt. Ltd. for their endless encouragement and support. My studies would not have been accomplished without their fantastic support. Finally, I want to thank my parents and friends for their relentless emotional support during my course of study...

Contents

List of Figures

List of Tables

Abbreviations

SER	**S**peech **E**motion **R**ecognition
CNN	**C**onvolutional **N**eural **N**etwork
MLP	**M**utli **L**ayer **P**erceptron
LSTM	**L**ong **S**hort **T**erm **M**emory
RNN	**R**eccurent **N**eural **N**etwork
VAE	**V**ariational **A**uto **E**ncoder
DeAE	**De**noising **A**uto **E**ncoder
DAE	**D**eep **A**uto **E**ncoder
AE	**A**uto **E**ncoder
CAE	**C**onvolutional **A**uto **E**ncoder
STFT	**S**hort **T**ime **F**ourier **T**ransform
MFCC	**M**el **F**requency **C**epstral **C**oefficients

Chapter 1

Introduction

1.1 Sound and Emotion

Sounds have been a phenomenal source of communication since multicellular animal life has begun its communication on the planet. Humans have always leveraged sounds to communicate, whether information or emotions. Humans have endeavoured to master their creation by learning to speak, have a conversation, sing a song, or invent musical instruments, always striving for new tonalities and better systems.

Humans communicate through expressive gestures of emotions and sentiments identified through experience and knowledge. These emotions might be expressed verbally or via body language. Emotions are a natural aspect of human life and, among other things, have a significant influence on decision making. The varieties of traits that may contain additional details well about the emotional significance of each utterance are examined in this research. The characteristics that contribute to emotions may differ between spoken languages[19].

In recent years, computer-aided emotion recognition has been a subject of attention. An effective human emotion detection algorithm will aid in making human-computer interaction more natural and pleasant. It has numerous uses in education, entertainment, customer service, etc. As two significant indicators of an effective human state, speech and facial expression play essential roles in emotion recognition[30]. Individuals express their emotions in a variety of ways. Recognizing emotions is a challenging problem because human emotions lack temporal constraints and are diverse[2].

1.2 Significance of Emotion Tagging

Emotions play a critical role in daily human interactions. The sentiment present in the sentences' utterances can change the entire meaning of the uttered sentence. For example, a sentence like "I am hungry!" can be interpreted differently when the tone of utterance changes; the sentence could be interpreted as the person feeling hungry when the utterance is usual. Still, the same can mean the person is not hungry when the tone of the phrase is sarcastic. Often the change in the meaning of the sentence can be observed with audio or visual cues. Furthermore, as humans already have past experiences, the human brain can process this information naturally.

Nevertheless, when the same behaviour has to be mimicked on a computer, the task becomes difficult because a computer cannot inherently distinguish emotions. Hence it becomes essential to train the computer with suitable algorithms and tagged emotions to perform the task of emotion classification.

The automatic recognition of spontaneous emotions from the speech is a challenging task. On the one hand, acoustic features need to be robust enough to capture the emotional content for various speaking styles. On the other, machine learning algorithms need to be insensitive to outliers while modelling the context.

Several game-changing advances have been observed, since the advent of deep neural networks in the last decade, in several established pattern recognition areas, such as an object, speech, and speaker recognition, as well as in combined problem-solving approaches, such as audio-visual recognition and the relatively new field of para linguistics[28]. Emotion recognition can find applications in different domains. For example, call centres can use emotion recognition in call centres, where the goal is to detect the caller's emotional state and provide feedback for the quality of the service [6].

1.3 Weakly Supervised Learning

"Weakly Supervised Learning" or "Weak supervision" is a branch of machine learning in which noisy, restricted, or inaccurate sources are employed to give supervision signals for categorizing vast volumes of training data in a supervised learning scenario. This methodology ameliorates the burden of acquiring hand-labelled data sets, which can be expensive or onerous. Instead, low-cost weak labels are applied with the awareness that they are imprecise but can still be used to build a powerful prediction model.

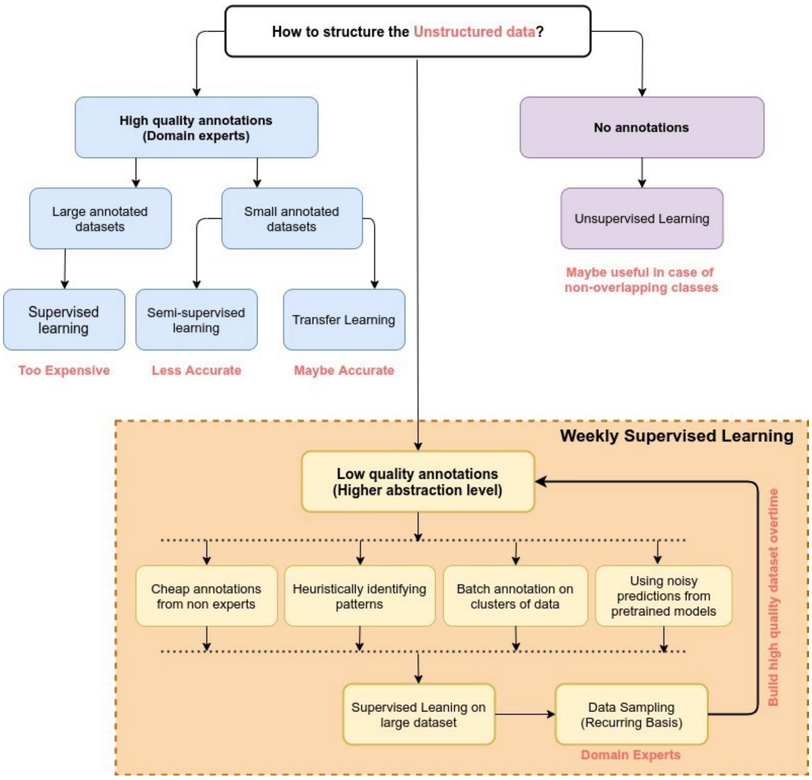
How to structure the Unstructured data?
High quality annotations (Domain experts)
No annotations
Large annotated datasets
Small annotated datasets
Unsupervised Learning
Maybe useful in case of non-overlapping classes
Supervised learning
Semi-supervised learning
Transfer Learning
Too Expensive
Less Accurate
Maybe Accurate
Weekly Supervised Learning
Low quality annotations (Higher abstraction level)
Cheap annotations from non experts
Heuristically identifying patterns
Batch annotation on clusters of data
Using noisy predictions from pretrained models
Supervised Leaning on large dataset
Data Sampling (Recurring Basis)
Domain Experts
Build high quality dataset overtime

FIGURE 1.1: Weakly Supervised Learning.

1.3.1 The problem of labelled training data

Machine learning methods and techniques have become more accessible to researchers and developers. However, its real-world efficacy is dependent on the availability of high-quality labelled training data. The necessity for labelled training data is typically a substantial impediment to the adoption of machine learning models This bottleneck effect manifests itself in a multitude of ways, as illustrated by the following examples.

Insufficient quantity of labelled data:

When machine learning techniques have been implemented in new applications or sectors, there is frequently inadequate training data to use standard approaches[24]. Some sectors have access to decades' worth of training data; those that do not are significantly disadvantaged. Obtaining training data in such cases may be inconvenient, costly, or impossible without patiently waiting for it to accumulate[33].

Insufficient subject-matter expertise to label data:

When labelling training data necessitates specialized skills, creating a good training data set can soon become prohibitively expensive [24]. This problem is expected to arise in machine learning applications[33].

Insufficient time to label and prepare data:

Most of the time spent on machine learning implementation is spent on data set preparation[24]. When a business or research sector works with challenges that, by definition, evolve quickly, it can be challenging to collect and process data quickly enough for conclusions to be helpful in real-world applications [33].

1.3.2 Types of weak labels:

Weak labels are designed to reduce the cost and improve the efficiency of human labour spent on hand-labeling data. They can take numerous shapes and can be divided into three types., Weak labels are designed to reduce the cost and improve the efficiency of human labour spent on hand-labelling data. They can take numerous shapes and can be divided into three types: There are three categories of inadequate supervision. The first is incomplete supervision, in which only a (typically small) portion of training data is labelled, and the rest is left unlabeled. This happens in a variety of tasks. For example, human annotators provide ground-truth labels; it is simple to obtain many photos via the Internet. However, only a limited portion of images can be annotated due to the human cost. The second kind is inexact supervision, which provides only coarse-grained labelling. Take a look at the picture categorisation challenge once again. Every item in the photographs should be annotated; however, we frequently only have image-level labels rather than object-level labels. The third category is incorrect supervision, which means that the labels supplied are not always correct. This can happen when the picture annotator is sloppy or tired or when specific photographs are difficult to classify. Weakly supervised learning is a catch-all name for a range of research that seeks to build predictive models by learning with little supervision.

1.3.3 Applications of weak supervision

In 2014, UC Berkeley researchers used weak supervision principles to develop an iterative learning algorithm that relies exclusively on labels generated by heuristics and eliminates the necessity for ground-truth labels[14]. The algorithm was applied to meter reading data to learn about a household's occupancy without ever asking for the data.

In 2018, UC Riverside researchers proposed a technique for localising actions/events in movies with only weak supervision, i.e., video-level labels, without any information on the start and end time of the events while training[22]. Their research presented an attention-based similarity between two films that serve as a regulariser for learning with imperfect labels. In 2019, they introduced a new problem of event localisation in videos employing text queries from users but with weak annotations while training[17].

1.4 Structure of thesis

This section briefly summarises the rest of the chapters, outlining the dissertation.

Chapter-2 : Literature Survey

This chapter will describe the previous works done in this area. We'll first go through the previous attempts to emotion tagging in audio signals and their results

Chapter-3 : The Data

This chapter will describe the methods used for extracting features from the audio signal and data sets that are used in this work.

Chapter-4 : Experiments

This chapter will describe the model architectures developed as part of this work.

Chapter-5 : Future Work

This chapter talks about the future scope to extend this work.

Chapter 2

Literature Survey

Humans utilise speech as one of their primary modes of communication. Because emotions are a means of communication for people, they are naturally employed in everyday speech to communicate their thoughts. Speech comprises information that is both linguistic and non-linguistic. Efficient communication via language and voice has permitted the exchange of ideas, messages, and perceptions. There are two essential aspects in voice-based signals: acoustic fluctuation and spoken words. Acoustic aspects of the speech signal, such as pitch, timing, voice quality, and articulation, are closely correlated with the underlying emotion due to the impacts of arousal in the neurological system, increased heart rate, etc. Emotion recognition in speech is based on the fluctuation of these qualities.

One of the significant issues in human-computer interaction is emotion identification from speech. The development of intelligent emotion identification systems is thus beneficial. The main objective of an emotion recognition system is to mimic human perception in the same manner as people recognise emotions such as anger, sadness, and happiness when conversing (Basu et al., 2017)[3]. Despite substantial study in emotion identification from voice, there are still various hurdles such as defective databases, low quality of recorded sounds, cross-database performance, and difficulties with speaker-independent recognition because everyone speaks differently. In the instance of call centre communications, the goal of such systems would be to determine the caller's emotional state and urgency (Bojani et al. 2020)[5]. And improve the functionality of call centres, particularly those providing health care assistance to the elderly and emergency call centres.

Many SER research procedures are founded on two distinct classification systems. The first involves the use of traditional classifiers such as SVM and artificial neural networks (ANN), while the second involves the use of deep learning classifiers CNN and DNN (Akçay and Ouz 2020)[1].

Zhang et al. (2016) [35] implemented four binary classification models:

- the basic model
- the single task (ST) model
- the multi-task feature selection/learning (MTFS/MTFL) model
- the group multi-task feature selection/learning (GMTFS/GMTFL) model

Four models were used for each emotion classification after extracting low-level acoustic descriptors (LLDs). It was evaluated on the RAVDESS dataset and obtained a maximum accuracy of 64.29%

Mustaqeem et al.[18] have proposed a Deep Stride CNN Architecture architecture encouraged by the idea of plain nets are specially designed for computer vision problems, like image classification, localization, tracking, and recognition to secure high-level accuracy. The research utilises the spectrograms generated for the audio signal and applies 2D convolution for audio emotion recognition.

MFCCs, Spectral Centroids, Delta and Delta–Delta MFCCs, as well as a bagging ensemble using SVM as a classifier, were utilised for speech identification on three different datasets: IITKGP-SEHSC, RAVDESS, and Berlin EMO-DB. Using the suggested technique, 75.69 percent accuracy was attained on the RAVDESS dataset (Bhavan et al. 2019)[4].

Another work, Tomba et al. (2018) [27], attempted to determine stress in speech using mean energy, mean intensity, and MFCC characteristics. Accuracy of 78.75 per cent and 89.16 per cent were attained using SVM and neural networks on the RAVDESS dataset, respectively.

In Deb and Dandapat (2016)[8], a relatively new feature, residual sinusoidal peak amplitude (RSPA), has been used for emotion classification. Using a sinusoidal model, the RSPA feature is calculated from the LP residual of the speech signal. The SVM classifier was employed again and assessed on the EMO-DB dataset, yielding a maximum accuracy of 74.4 per cent.

Similarly, architectures such as convolutional neural network (CNN) and long short-term memory (LSTM) have been used to assess the emotion capture capabilities of several conventional speech representations such as Mel spectrogram, magnitude spectrogram, and Mel-frequency Cepstral Coefficients (MFCC's). Pandey et al. [21] employed a bidirectional long short term memory network and a convolutional neural network, and the best accuracy was 82.35 per cent for CNN + BLSTM architecture using MFCC as input for EMODB. Jannat et al. [13] examined a convolutional neural network model using RAVDESS, although the accuracy of the solitary audio tests is relatively low at 66.41 per cent.

One 1D CNN LSTM and one 2D CNN LSTM were built by Zhao et al. [37] to train on local and global emotion-related characteristics from speech and log-Mel spectrograms, respectively. Accuracy rates of 95.33 percent and 95.89 percent for speaker-dependent and speaker-independent trials in Berlin EmoDB, and 89.16 percent and 52.14 percent for speaker-dependent and speaker-independent experiments in the IEMOCAP database, respectively, were achieved.

Deng et al.[10] suggested a sparse autoencoder approach for feature transfer learning for voice emotion identification . The datasets had an average accuracy of 51.6 percent (original) and 59.9 percent (reconstructed). Deng et al.[9] proposed the semi-supervised autoencoder (SS-AE) to learn from labelled and unlabeled data. It improves on a well-known unsupervised deep denoising autoencoder. SS-AE-Skip, a variation of SS-AE that introduces skip connections from the bottom layer to the higher layer, was also implemented. The average UAR for SS-AE and SS-AE-Skip is 42.7 percent and 42.8 percent, respectively.

Chapter 3

The Data

3.1 Data Collection and Exploratory Data Analysis

Procuring the data sets has been a difficult task as many data sets are not available relevant to the Audio emotion tagging. Hence the datasets TESS, RAVDESS, SAVEE and CREMA-D have been used in this work because of their availability. The next sections will describe about the datasets and their features.

3.1.1 Datasets

3.1.1.1 Surrey Audio-Visual Expressed Emotion [SAVEE]

This dataset[12] is provided by researchers at the University of Surrey, Guildford, England. The speakers are aged from 27 to 31 years. Emotion has been described psychologically in discrete categories: anger, disgust, fear, happiness, sadness and surprise. The audio files from this dataset have european accent. The database was captured in CVSSP's 3D vision laboratory over several months during different time period of the year from four actors.

3.1.1.2 Ryerson AV Database of Emotional Speech and Song [RAVDESS]

The database[16] 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. Each expression is produced at two levels of emotional intensity (normal, strong), with an additional neutral expression.

3.1.1.3 Toronto emotional speech set [TESS]

These stimuli were modeled on the Northwestern University Auditory Test No. 6 (NU-6; Tillman and Carhart, 1966). A set of 200 target words were spoken in the carrier phrase "Say the word ____" by two actresses (aged 26 and 64 years) and recordings were made of the set portraying each of seven emotions (anger, disgust, fear, happiness, pleasant surprise, sadness, and neutral). There are 2800 stimuli in total. Two actresses were recruited from the Toronto area. Both actresses speak English as their first language, are university educated, and have musical training. Audio metric testing indicated that both actresses have thresholds within the normal range [23].

3.1.1.4 Crowd-sourced Emotional Multimodal Actors Dataset [CREMA-D]

CREMA-D is a data set of 7,442 original clips from 91 actors. These clips were from 48 male and 43 female actors between the ages of 20 and 74 coming from a variety of races and ethnicities (African America, Asian, Caucasian, Hispanic, and Unspecified). Actors spoke from a selection of 12 sentences. The sentences were presented using one of six different emotions (Anger, Disgust, Fear, Happy, Neutral and Sad)[7].

The Audio data available in the datasets is mere voice recording, but for a neural network to understand the data it should be in numbers so, I have used below techniques to draw some data from inferences of the data in frequency domain instead of time domain, because analyzing audio signals in frequency domain is easier and advanced math techniques like Fast Fourier transforms for frequency domain analysis.

All four datasets are good. After listening to them and running some very preliminary inspections, I believe we can incorporate all of them so that we can minimise overfitting issues. One of the challenges I've noticed with many other researches that attempted to build an emotion classifier is that they prefer to adhere to a single dataset. And, while their hold-out set accuracy is great, they do not do well on fresh, previously unseen dataset.

This is attributed to the reason that the classifier is trained on the same dataset, and given the similar circumstances under which the dataset was generated or produced (eg. audio quality, speaker repetition, duration and sentence uttered). To allow it to perform effectively on fresh datasets, it must be subjected to noise, which forces it to work hard to identify the true differentiating aspects of the emotion.

As we have only the audio files it makes it difficult to the model to parse it and we need to get them into right mathematical format for obtaining meaningful characteristics from audio for the classifier.

3.2 Feature Selection

There are two category of features:

Time domain characteristics These are easier to extract and comprehend, such as signal energy, zero crossing rate, maximum amplitude, lowest energy, and so on.

Frequency domain characteristics By translating the time-based signal into the frequency domain, frequency-based characteristics may be acquired. While they are more difficult to understand, they contain additional information that can be very useful, such as pitch, rhythms, melody, and so on.

3.2.1 Mel scale

The mel scale is a perceptive scale of pitches believed to be equal in distance from one another by listeners. The reference point between this scale and regular frequency measurement is determined by giving a 1000 Hz tone a perceptual pitch of 1000 mels, 40 dB over the listener's threshold. Listeners judge increasingly large intervals above about 500 Hz to produce equal pitch increments[31].

$$m = 2595 \log_{10} \left(1 + \frac{f}{700}\right) \tag{3.1}$$

3.2.2 MFCC

It is an abbreviation for Mel-frequency cepstral coefficient, and it is a decent "representation" of the vocal tract that generates the sound. Cepstrum is the rate of change information in spectral bands. In conventional temporal signal analysis, any periodic component (for example, echoes)

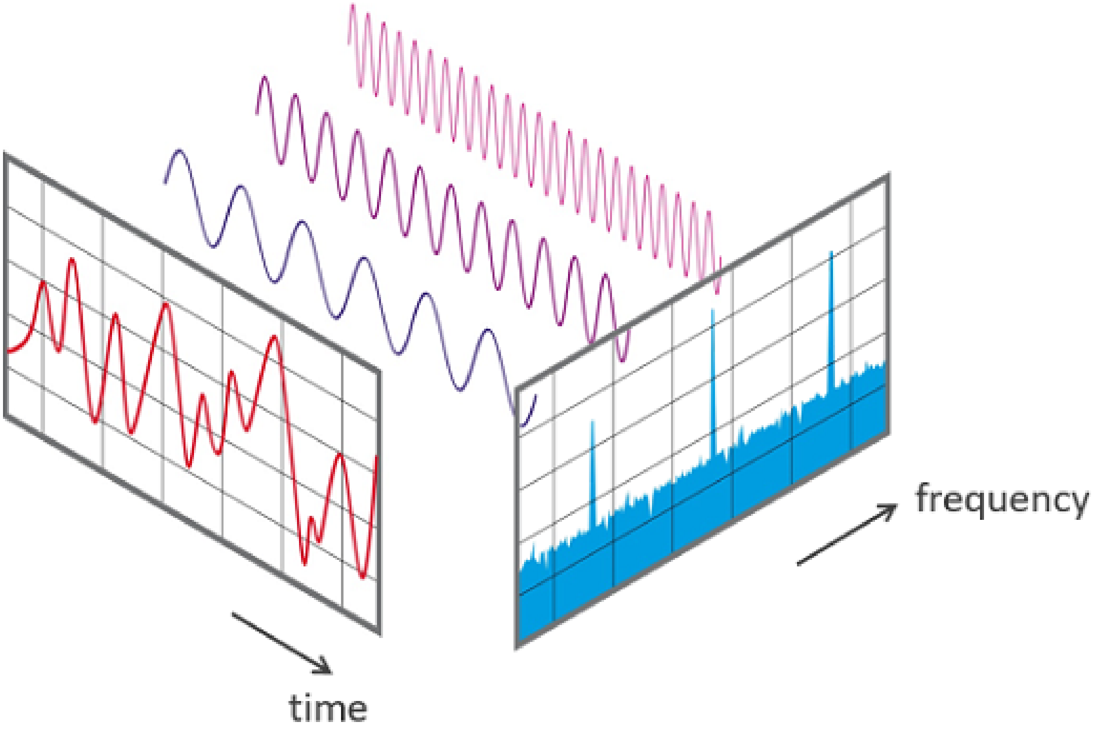

Figure 3.1: Analysis of Audio Signal in Time and Frequency Domain.

appears as sharp peaks in the associated frequency spectrum. This is achieved by performing a Fourier transform on the time signal.

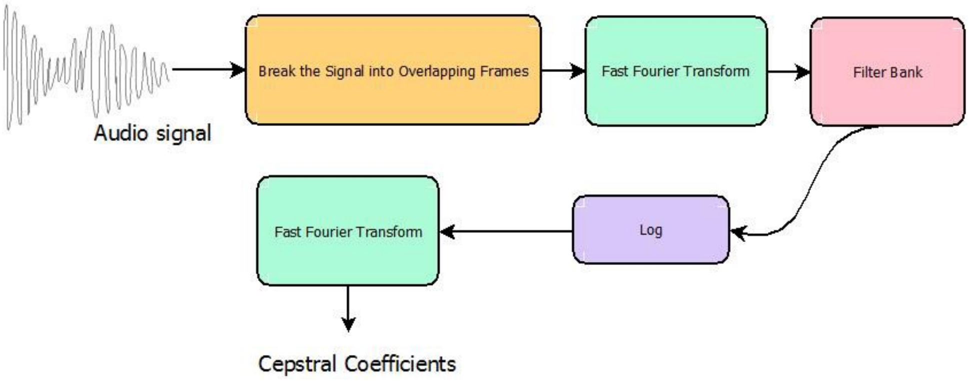

Figure 3.2: Mel-frequency cepstral coefficients

There is a noticeable difference between male and female utterances of the identical sentence in that females tend to have a higher pitch.

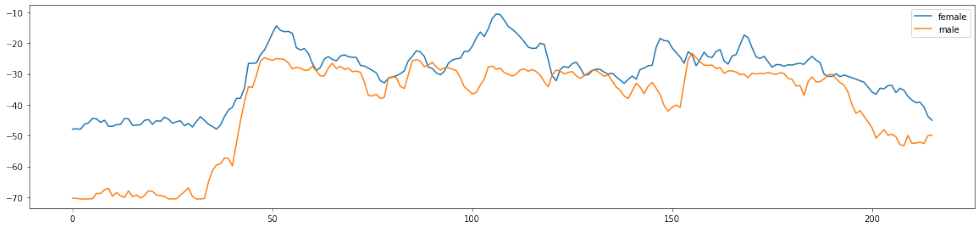

FIGURE 3.3: Pitch difference between male and female utterances with Angry emotion from RAVDESS dataset

3.2.3 Zero Crossing Rate

The zero-crossing rate (ZCR) is the rate at which a signal changes from positive to zero to negative or from negative to zero to positive. Its value has been widely used in both speech recognition and music information retrieval, being a key feature to classify percussive sounds[5].

ZCR is defined formally as

$$zcr = \frac{1}{T-1}\sum_{t=1}^{T-1} \nVdash_{\mathbb{R}_{<0}}(s_t s_{t-1}) \tag{3.2}$$

where s is a signal of length T and $\nVdash_{\mathbb{R}_{<0}}$ is an indicator function.

In some cases only the “positive-going” or “negative-going” crossings are counted, rather than all the crossings, since between a pair of adjacent positive zero-crossings there must be a single negative zero-crossing.

For monophonic tonal signals, the zero-crossing rate can be used as a primitive pitch detection algorithm. Zero crossing rates are also used for Voice activity detection (VAD), which determines whether human speech is present in an audio segment or not[34].

3.2.4 Chroma STFT

An audio’s Chroma value essentially represents the strength of the twelve various pitch classes used to study music. They may be used to differentiate the pitch class profiles of audio streams. STFT[1] encodes information concerning pitch categorization and signal structure. It represents the spike as a series of high and low numbers.

[1] Short-time Fourier transform (STFT), is a Fourier-related transform used to determine the sinusoidal frequency and phase content of local sections of a signal as it changes over time[25].

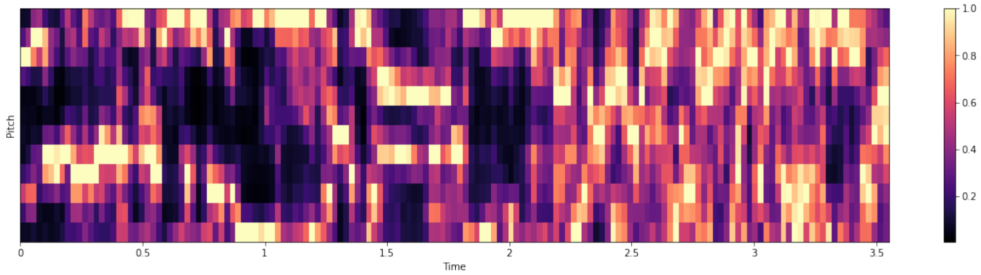

Figure 3.4: ChromaSTFT of an Angry emotion from CREMA−D dataset

3.3 Augmentation

Data augmentation in data analysis are techniques used to increase the amount of data by adding slightly modified copies of already existing data or newly created synthetic data from existing data. It acts as a regularizer and helps reduce overfitting when training a machine learning model. It is closely related to oversampling in data analysis[26].

In order for the model to be resilient to unknown data I have applied the following augmentation techniques. Added static noise in the background, phase shift, signal stretch, pitch, dynamic change, speed and pitch.

Chapter 4

Experiments

4.1 Exploring Model Architectures

I carried out experiments to on 1D, 2D CNN architectures and AE+CNN architecure. I'll discuss the architecture and the observations of the experiments. The suggested approach is tested using Four datasets: Ravdess, CREMA-D and SAVEE were used for training and TESS dataset was used for testing. MFCC and Data augmentation techniques are used for feature extraction from raw audio files.

4.1.1 Convolutional Neural Network

A convolutional neural network (CNN, or ConvNet) is a type of artificial neural network that is extensively used in deep learning to interpret visual information[29]. They are also known as shift invariant or space invariant artificial neural networks (SIANN), since they are built on the shared-weight architecture of convolution kernels or filters that slide along input features and give translation equivariant outputs known as feature maps[36].

4.1.1.1 1D-Convolutional Neural Network

In this architecture seven 1D convolutional layers followed by a max pooling layer and drop out layers was used.

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv1d_16 (Conv1D)          (None, 216, 256)          2304

 activation_18 (Activation)  (None, 216, 256)          0

 conv1d_17 (Conv1D)          (None, 216, 256)          524544

 batch_normalization_4 (Batc  (None, 216, 256)         1024
 hNormalization)

 activation_19 (Activation)  (None, 216, 256)          0

 dropout_4 (Dropout)         (None, 216, 256)          0

 max_pooling1d_4 (MaxPooling  (None, 27, 256)          0
 1D)

 conv1d_18 (Conv1D)          (None, 27, 128)           262272

 activation_20 (Activation)  (None, 27, 128)           0

 conv1d_19 (Conv1D)          (None, 27, 128)           131200

 activation_21 (Activation)  (None, 27, 128)           0

 conv1d_20 (Conv1D)          (None, 27, 128)           131200

 activation_22 (Activation)  (None, 27, 128)           0

 conv1d_21 (Conv1D)          (None, 27, 128)           131200

 batch_normalization_5 (Batc  (None, 27, 128)          512
 hNormalization)

 activation_23 (Activation)  (None, 27, 128)           0

 dropout_5 (Dropout)         (None, 27, 128)           0

 max_pooling1d_5 (MaxPooling  (None, 3, 128)           0
 1D)

 conv1d_22 (Conv1D)          (None, 3, 64)             65600

 activation_24 (Activation)  (None, 3, 64)             0
 conv1d_23 (Conv1D)          (None, 3, 64)             32832

 activation_25 (Activation)  (None, 3, 64)             0

 flatten_2 (Flatten)         (None, 192)               0

 dense_2 (Dense)             (None, 7)                 1351

 activation_26 (Activation)  (None, 7)                 0

=================================================================
Total params: 1,284,039
Trainable params: 1,283,271
Non-trainable params: 768
_________________________________________________________________
```

4.1.1.2 2D-Convolutional Neural Network

The 2D CNN receives input data in the form of a 2D array of 30 MFCC bands by 216 audio length. Similar to mimic the scenario of usinf as a 30 x 216 pixel image. It has four convolution blocks: batch normalisation, max pooling, and a dropout node. And Adam is our optimizer.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_5 (InputLayer)         (None, 30, 216, 1)        0
_________________________________________________________________
conv2d_17 (Conv2D)           (None, 30, 216, 32)       1312
_________________________________________________________________
batch_normalization_21 (Batc (None, 30, 216, 32)       128
_________________________________________________________________
activation_21 (Activation)   (None, 30, 216, 32)       0
_________________________________________________________________
max_pooling2d_17 (MaxPooling (None, 15, 108, 32)       0
_________________________________________________________________
dropout_25 (Dropout)         (None, 15, 108, 32)       0
_________________________________________________________________
conv2d_18 (Conv2D)           (None, 15, 108, 32)       40992
_________________________________________________________________
batch_normalization_22 (Batc (None, 15, 108, 32)       128
_________________________________________________________________
activation_22 (Activation)   (None, 15, 108, 32)       0
_________________________________________________________________
max_pooling2d_18 (MaxPooling (None, 7, 54, 32)         0
_________________________________________________________________
dropout_26 (Dropout)         (None , 7, 54, 32)        0

_________________________________________________________________
conv2d_19 (Conv2D)           (None , 7, 54, 32)        40992

_________________________________________________________________
batch_normalization_23 (Batc (None , 7, 54, 32)        128

_________________________________________________________________
activation_23 (Activation)   (None , 7, 54, 32)        0

_________________________________________________________________
max_pooling2d_19 (MaxPooling (None , 3, 27, 32)        0

_________________________________________________________________
dropout_27 (Dropout)         (None , 3, 27, 32)        0

_________________________________________________________________
conv2d_20 (Conv2D)           (None , 3, 27, 32)        40992

_________________________________________________________________
batch_normalization_24 (Batc (None , 3, 27, 32)        128

_________________________________________________________________
activation_24 (Activation)   (None , 3, 27, 32)        0

_________________________________________________________________
max_pooling2d_20 (MaxPooling (None , 1, 13, 32)        0

_________________________________________________________________
dropout_28 (Dropout)         (None , 1, 13, 32)        0

_________________________________________________________________
flatten_5 (Flatten)          (None , 416)              0

_________________________________________________________________
dense_9 (Dense)              (None , 64)               26688

_________________________________________________________________
dropout_29 (Dropout)         (None , 64)               0

_________________________________________________________________
batch_normalization_25 (Batc (None , 64)               256

_________________________________________________________________
activation_25 (Activation)   (None , 64)               0

_________________________________________________________________
dropout_30 (Dropout)         (None , 64)               0

_________________________________________________________________
dense_10 (Dense)             (None , 7)                455
=================================================================
Total params: 152,199

Trainable params: 151,815

Non-trainable params: 384

_________________________________________________________________
```

4.1.2 Convolutional Auto Encoder and Convolutional Neural Network

I have employed a basic convolutional autoencoder. For the three datasets, the suggested autoencoder architecture is the same. The encoding dimension used is 64, and the input shape is (128,). The encoder and decoder layers have been combined into a single layer, with encoded label holding the encoded representation of the input and 'decoded' label storing the lossy reconstruction of the input. A separate encoder and decoder model is also constructed. Now, in order to train our autoencoder to reconstruct the audio flow, the developed model is first configured to use categorical cross-entropy as the loss function and the SGD optimizer to optimise the loss function. After transferring the encoded input to the decoder layer, the decoded input has the same size as the original input but a lower pixel value. As a result, we will obtain an output with form (128) but reduced dimensions. The reconstructed input accuracies after using our proposed autoencoder model in RAVDESS,CREMA-D and SAVEE datasets is 85.92%. The Encoder model is now connected to a CNN classifier to identify the emotions.

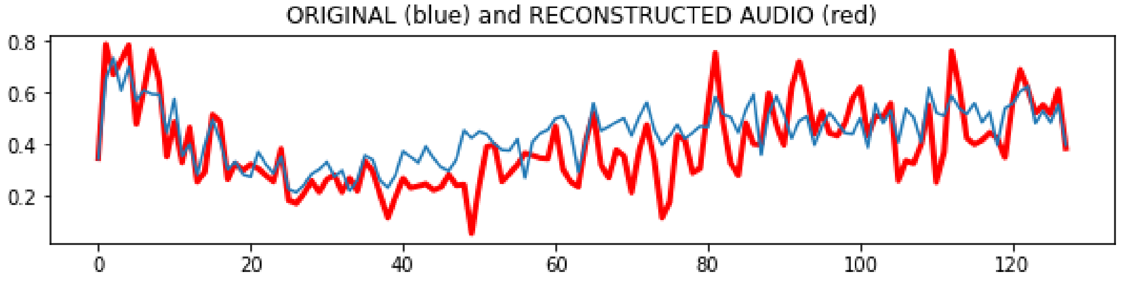

FIGURE 4.1: Reconstructed Audio.

CAE + CNN :

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
Input (InputLayer)           (None , 216 , 1)          0
_________________________________________________________________
line1 (Conv1D)               (None , 216 , 256)        2304
_________________________________________________________________
line2 (MaxPooling1D)         (None , 36 , 256)         0
_________________________________________________________________
line3 (Conv1D)               (None , 36 , 128)         262272
_________________________________________________________________
line4 (MaxPooling1D)         (None , 6 , 128)          0
_________________________________________________________________
```

```
line5 (Conv1D)               (None, 6, 64)           65600
_________________________________________________________________
line6 (MaxPooling1D)         (None, 1, 64)           0
_________________________________________________________________
conv1d_80 (Conv1D)           (None, 1, 128)          49280
_________________________________________________________________
conv1d_81 (Conv1D)           (None, 1, 64)           49216
_________________________________________________________________
conv1d_82 (Conv1D)           (None, 1, 32)           12320
_________________________________________________________________
activation_84 (Activation)   (None, 1, 32)           0
_________________________________________________________________
conv1d_83 (Conv1D)           (None, 1, 16)           3088
_________________________________________________________________
flatten_11 (Flatten)         (None, 16)              0
_________________________________________________________________
dense_11 (Dense)             (None, 7)               119
_________________________________________________________________
activation_85 (Activation)   (None, 7)               0
=================================================================
Total params: 444,199
Trainable params: 444,199
Non-trainable params: 0
_________________________________________________________________
```

4.2 Conclusion

The CNN architectures have given decent accuracies on unseen test data. The 1D CNN architecture has given an accuracy of 50% on test data, and the 2D CNN architecture has given 66% accuracy on test data. The Convolutional Auto encoder and Convolutional Neural Network Architecture has given 53% test accuracy. From the results its evident that when the Speech emotion recognition models have a bias towards the speaker voice characteristics and the model tends to over fit when deeper architectures are used with speakers from different nationalities. The model performs well with shallow architectures. Even the results using dimensionality reduction techniques as Auto encoders provide higher accuracies, the encoders could be used to clear out unneeded noise in the data. The paradigm of weak supervision has been fruitful and has helped in yielding decent results in classifying the emotions Neutral, Happy, Fear, Surprise, Sad, Disgust and Angry. But however there has been an overlap when recognising the emotions Happy - Surprise, Disgust - Angry, Sad-Fear. The accuracy of the models improve when they are used to predict genders but when the audio data set is analysed only for emotions the overlap percentage increases.

Chapter 5

Future Work

The results from the work show promising results in using CNN and Auto encoder architectures with weak supervision, hence these could be explored deeper. Architectures like LSTM, RNN also could be explored. The decrease in accuracy is because the data set is divided into two genders, hence more complex architectures which aren't biased towards gender could be explored. Different encoders, even in combination, such as de-noising AE, are likely to enhance the results when used along with weak supervision. Also datasets with more emotions could be used along with weak supervision to improve the model performance.

Appendix A

Experimental Results

The results obtained from multiple architectures is as follows. The datasets CREMA-D, RAVDESS, SAVEE were used for training the model and the dataset TESS was used to benchmark the model.

A.1 CNN Architectures

A.1.1 1D – Convolutional Neural Network

Classes	**Precision**	**Recall**	**F-1 score**
Angry	0.67	0.43	0.52
Disgust	0.48	0.49	0.48
Fear	0.59	0.3	0.4
Happy	0.42	0.49	0.45
Neutral	0.38	0.72	0.5
Sad	0.56	0.51	0.53
Surprise	0.93	0.58	0.71
Macro average	0.58	0.5	0.52
Weighted average	0.54	0.5	0.5

Table A.1: Classification report of 1D CNN Architecture

The accuracy of the model during testing is 50%. The model is compiled using SGD optimiser as it has yielded the best results than RMSprop and Adam.

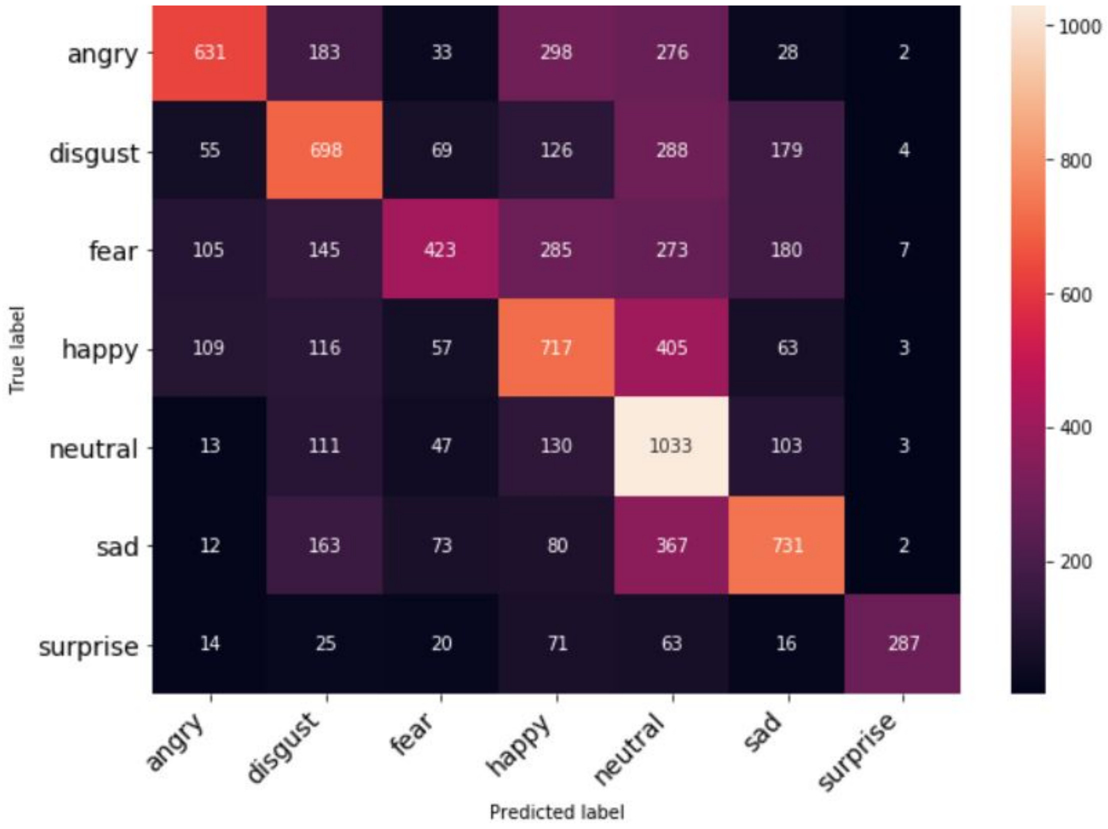

Figure A.1: Confusion Matrix for 1D-CNN Architecture

A.1.2 2D – Convolutional Neural Network

Classes	Precision	Recall	F-1 score
Angry	0.67	0.85	0.75
Disgust	0.71	0.57	0.63
Fear	0.58	0.57	0.57
Happy	0.65	0.63	0.64
Neutral	0.66	0.77	0.71
Sad	0.65	0.56	0.6
Surprise	0.93	0.78	0.85
Macro average	0.69	0.68	0.68
Weighted average	0.67	0.66	0.66

Table A.2: Classification report for 2D CNN Architecture

The accuracy of the model during testing is 66%.The model is compiled using Adam optimiser as it has yielded the best results than RMSprop and SGD.

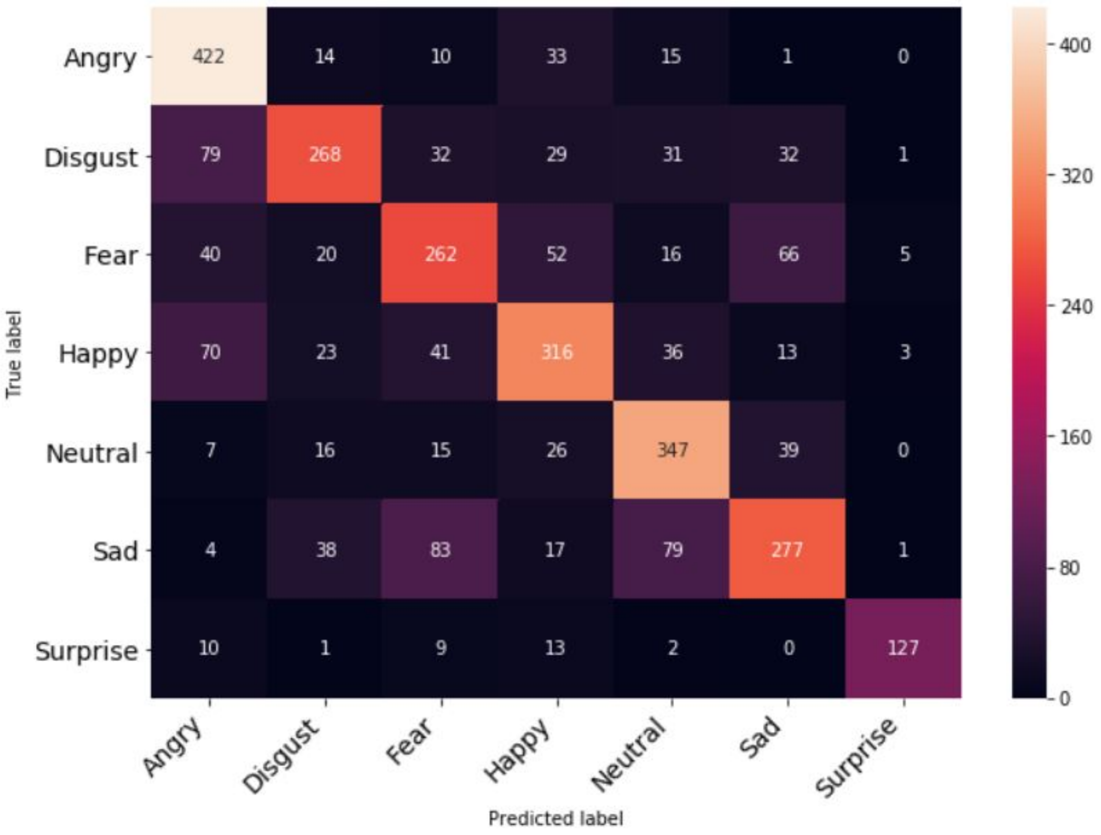

FIGURE A.2: Confusion Matrix for 2D CNN Architecture

A.2 Convolutional Auto Encoder and Convolutional Neural Network

Classes	Precision	Recall	F-1 score
Angry	0.73	0.45	0.55
Disgust	0.47	0.48	0.47
Fear	0.47	0.45	0.46
Happy	0.44	0.5	0.47
Neutral	0.59	0.54	0.56
Sad	0.49	0.69	0.58
Surprise	0.75	0.68	0.71
Macro average	0.56	0.54	0.54
Weighted average	0.54	0.53	0.53

TABLE A.3: Classification report for CAE-CNN Architecture

The accuracy of the model during testing is 53%. The model is compiled using RMSprop optimiser as it has yielded the best results than SGD and Adam.

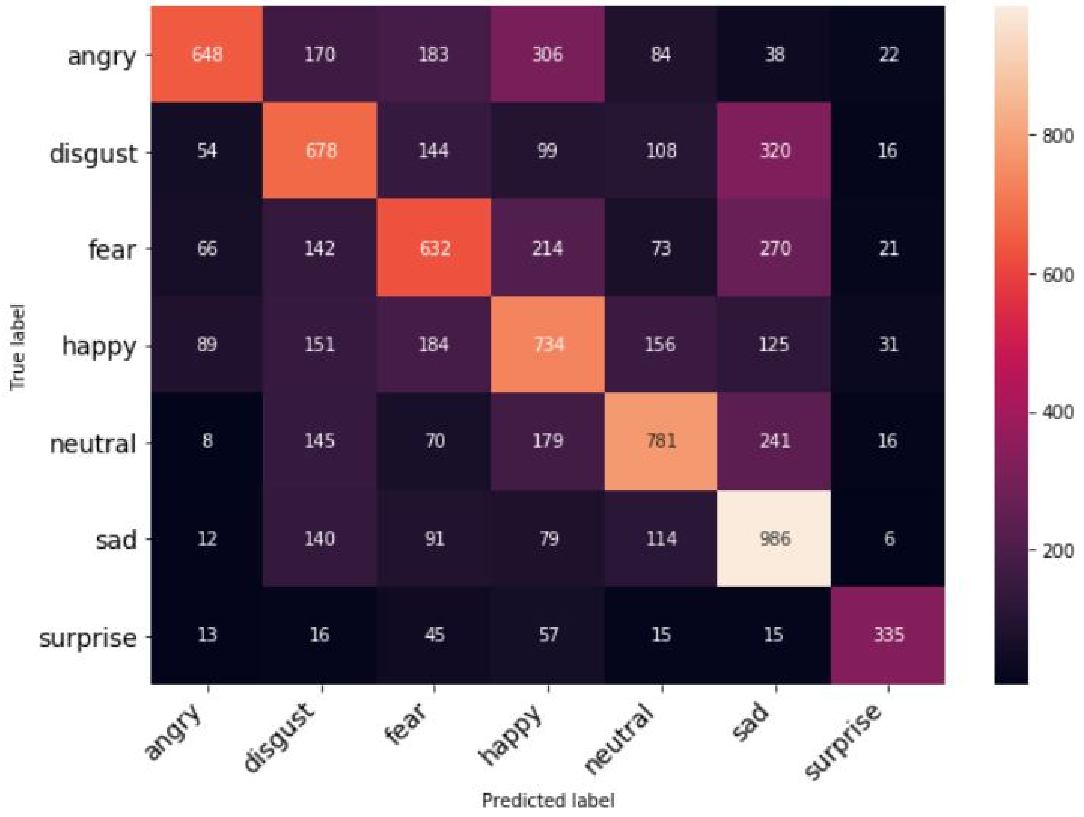

Figure A.3: Confusion Matrix for CAE-CNN Architecture

Appendix B

Mathematical Concepts

B.1 Categorical Cross entropy

Categorical cross entropy is a loss function that is used in multi-class classification tasks. These are tasks where an example can only belong to one out of many possible categories, and the model must decide which one. Formally, it is designed to quantify the difference between two probability distributions. The categorical cross entropy loss function calculates the loss of an example by computing the following sum:

$$\text{Loss} = -\sum_{i=1}^{\substack{\text{output}\\\text{size}}} y_i \cdot \log \hat{y}_i$$

where $\hat{y}_i$ is the i-th scalar value in the model output, y_i is the corresponding target value, and output size is the number of scalar values in the model output. This loss is an excellent indicator of how different two discrete probability distributions are from one another. In this context, y_i represents the chance that event i happens, and the total of all y_i equals 1, implying that only one event is possible. The negative sign assures that the loss decreases as the distributions approach each other.

B.2 Optimising using Stochastic Gradient Descent

Stochastic gradient descent (SGD) is an iterative method for optimising an objective function with appropriate smoothness constraints (e.g. differential or subdifferentiable). It is a stochastic approximation to gradient descent optimization because it substitutes the real gradient (derived from the whole data set) with an estimate of it (calculated from a randomly selected subset of the data). This minimises the computing cost, allowing for faster iterations in exchange for a lower convergence rate, especially in high-dimensional optimization problems. [32]

Let's suppose we want to fit a straight line $\hat{y} = w_1 + w_2 x \hat{y} = w_1 + w_2 x$ to a training set with observations $(x_1, x_2, \ldots, x_n)(x_1, x_2, \ldots, x_n)$ and corresponding estimated responses $(\hat{y_1}, \hat{y_2}, \ldots, \hat{y_n})$ using least squares. The objective function to be minimized is:

$$Q(w) = \sum_{i=1}^{n} Q_i(w) = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 = \sum_{i=1}^{n} (w_1 + w_2 x_i - y_i)^2 . \tag{B.1}$$

The last line in the above pseudocode for this specific problem will become:

$$\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} := \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} - \eta \begin{bmatrix} \frac{\partial}{\partial w_1}(w_1 + w_2 x_i - y_i)^2 \\ \frac{\partial}{\partial w_2}(w_1 + w_2 x_i - y_i)^2 \end{bmatrix} = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} - \eta \begin{bmatrix} 2(w_1 + w_2 x_i - y_i) \\ 2x_i(w_1 + w_2 x_i - y_i) \end{bmatrix} . \tag{B.2}$$

Note that in each iteration (also called update), the gradient is only evaluated at a single point x_i instead of at the set of all samples. The main difference between this method and regular (Batch) Gradient Descent is that just one piece of data from the data set is utilised to compute the step, and the piece of data is chosen at random at each step.

B.2.1 RMSProp

RMSProp (for Root Mean Square Propagation) is also a method in which the learning rate is adapted for each of the parameters. The idea is to divide the learning rate for a weight by a running average of the magnitudes of recent gradients for that weight[20]. So, first the running average is calculated in terms of means square,

$$v(w, t) := \gamma v(w, t-1) + (1 - \gamma)(\nabla Q_i(w))^2 \tag{B.3}$$

where, γ is the forgetting factor. And the parameters are updated as,

$$w := w - \frac{\eta}{\sqrt{v(w, t)}} \nabla Q_i(w) \tag{B.4}$$

RMSProp has shown good adaptation of learning rate in different applications. RMSProp can be seen as a generalization of Rprop and is capable to work with mini-batches as well opposed to only full-batches[20].

B.2.2 Adam

Adam[15][32] (short for Adaptive Moment Estimation) is an update to the RMSProp optimizer. In this optimization algorithm, running averages of both the gradients and the second moments of the gradients are used. Given parameters $w^{(t)}$ and a loss function $L^{(t)}$, where t t indexes the current training iteration (indexed at 0), Adam's parameter update is given by:

$$m_w^{(t+1)} \leftarrow \beta_1 m_w^{(t)} + (1-\beta_1)\nabla_w L^{(t)} \tag{B.5}$$

$$v_w^{(t+1)} \leftarrow \beta_2 v_w^{(t)} + (1-\beta_2)(\nabla_w L^{(t)})^2 v_w^{(t+1)} \leftarrow \beta_2 v_w^{(t)} + (1-\beta_2)(\nabla_w L^{(t)})^2 \tag{B.6}$$

$$\hat{m}_w = \frac{m_w^{(t+1)}}{1-\beta_1} \tag{B.7}$$

$$\hat{v}_w = \frac{v_w^{(t+1)}}{1-\beta_2} \tag{B.8}$$

$$w^{(t+1)} \leftarrow w^{(t)} - \eta \frac{\hat{m}_w}{\sqrt{\hat{v}_w} + \epsilon} \tag{B.9}$$

where ϵ is a small scalar (e.g. 10^{-8}) used to prevent division by 0, and β_1 (e.g. 0.9) and β_2 (e.g. 0.999) are the forgetting factors for gradients and second moments of gradients, respectively. Squaring and square-rooting is done element-wise.

B.2.3 AdaGrad

AdaGrad (for adaptive gradient algorithm) is a modified stochastic gradient descent algorithm with per-parameter learning rate, first published in 2011[11]. Informally, this increases the learning rate for sparser parameters and decreases the learning rate for ones that are less sparse. This strategy often improves convergence performance over standard stochastic gradient descent in settings where data is sparse and sparse parameters are more informative. Examples of such applications include natural language processing and image recognition[11] It still has a base learning rate η, but this is multiplied with the elements of a vector Gj, j which is the diagonal of the outer product matrix

$$G = \sum_{\tau=1}^{t} g_\tau g_\tau^\mathsf{T} \tag{B.10}$$

where $g_\tau = \nabla Q_i(w) g_\tau = \nabla Q_i(w)$, the gradient, at iteration τ. The diagonal is given by

$$G_{j,j} = \sum_{\tau=1}^{t} g_{\tau,j}^2. \tag{B.11}$$

This vector is updated after every iteration. The formula for an update is now

$$w := w - \eta \, \mathrm{diag}(G)^{-\frac{1}{2}} \odot g \tag{B.12}$$

or, written as per-parameter updates,

$$w_j := w_j - \frac{\eta}{\sqrt{G_{j,j}}} g_j \tag{B.13}$$

Each $G(i,i)$ gives rise to a scaling factor for the learning rate that applies to a single parameter w_i. Since the denominator in this factor, $\sqrt{G_i} = \sqrt{\sum_{\tau=1}^{t} g_\tau^2}$ is the $l2$ norm of previous derivatives, extreme parameter updates get dampened, while parameters that get few or small updates receive higher learning rates.

Appendix C

Autoencoder

An autoencoder simulates the given training data's density function to reconstruct the input. An autoencoder consists of an encoder that maps the data x to the latent space z, and a decoder that maps a sample from z to reconstruct the output $\hat{\mathrm{x}}$. It is a feed-forward network with back-propagation used to reduce the L2 loss between the output and input.

$$\begin{tikzcd}
\hat{\mathrm{x}} \\
z \arrow[u] \\
x \arrow[u]
\end{tikzcd}$$

The latent space z must have a lower dimensional feature representation than the input space x in order for z to be pushed to capture significant causes of variation in the data via training. The autoencoder may be used to seed a supervised learning model with superior features learnt from the encoder. It is used to reduce dimensionality by learning a reduced dimensional representation of the latent variable z. The density function of the latent space z is not provided by the autoencoder. We can use an autoencoder to reconstruct, but we cannot sample from the latent distribution to generate fresh samples.

Bibliography

[1] Mehmet Berkehan Akçay and Kaya Oğuz. "Speech emotion recognition: Emotional models, databases, features, preprocessing methods, supporting modalities, and classifiers". In: *Speech Communication* 116 (2020), pp. 56–76. ISSN: 0167-6393. DOI: https://doi.org/10.1016/j.specom.2019.12.001. URL: https://www.sciencedirect.com/science/article/pii/S0167639319302262.

[2] Christos-Nikolaos Anagnostopoulos, Theodoros Iliou, and Ioannis Giannoukos. "Features and classifiers for emotion recognition from speech: a survey from 2000 to 2011". In: *Artificial Intelligence Review* 43.2 (2015), pp. 155–177. ISSN: 1573-7462. DOI: 10.1007/s10462-012-9368-5. URL: https://doi.org/10.1007/s10462-012-9368-5.

[3] Saikat Basu et al. "A review on emotion recognition using speech". In: Mar. 2017, pp. 109–114. DOI: 10.1109/ICICCT.2017.7975169.

[4] Anjali Bhavan et al. "Bagged support vector machines for emotion recognition from speech". In: *Knowledge-Based Systems* 184 (2019), p. 104886. ISSN: 0950-7051. DOI: https://doi.org/10.1016/j.knosys.2019.104886. URL: https://www.sciencedirect.com/science/article/pii/S0950705119303533.

[5] Milana Bojanic, Vlado Delić, and Alexey Karpov. "Call Redistribution for a Call Center Based on Speech Emotion Recognition". In: *Applied Sciences* 10 (July 2020), p. 4653. DOI: 10.3390/app10134653.

[6] F. Burkhardt et al. "Detecting anger in automated voice portal dialogs". English (US). In: *INTERSPEECH 2006 and 9th International Conference on Spoken Language Processing, INTERSPEECH 2006 - ICSLP*. Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH. INTERSPEECH 2006 and 9th International Conference on Spoken Language Processing, INTERSPEECH 2006

- ICSLP ; Conference date: 17-09-2006 Through 21-09-2006. 2006, pp. 1053–1056. ISBN: 9781604234497.

[7] Houwei Cao et al. "CREMA-D: Crowd-Sourced Emotional Multimodal Actors Dataset". In: *IEEE Transactions on Affective Computing* 5.4 (2014), pp. 377–390. DOI: 10.1109/TAFFC.2014.2336244.

[8] Suman Deb and S. Dandapat. "Emotion classification using residual sinusoidal peak amplitude". In: June 2016, pp. 1–5. DOI: 10.1109/SPCOM.2016.7746697.

[9] Jun Deng et al. "Semisupervised Autoencoders for Speech Emotion Recognition". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 26.1 (2018), pp. 31–43. DOI: 10.1109/TASLP.2017.2759338.

[10] Jun Deng et al. "Sparse Autoencoder-Based Feature Transfer Learning for Speech Emotion Recognition". In: *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction* (2013), pp. 511–516.

[11] John Duchi, Elad Hazan, and Yoram Singer. "Adaptive Subgradient Methods for Online Learning and Stochastic Optimization". In: *Journal of Machine Learning Research* 12.61 (2011), pp. 2121–2159. URL: http://jmlr.org/papers/v12/duchi11a.html.

[12] Philip Jackson and Sana ul haq. *Surrey Audio-Visual Expressed Emotion (SAVEE) database.* Apr. 2011.

[13] Rahatul Jannat et al. "Ubiquitous Emotion Recognition Using Audio and Video Data". In: *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers.* UbiComp '18. Singapore, Singapore: Association for Computing Machinery, 2018, 956–959. ISBN: 9781450359665. DOI: 10.1145/3267305.3267689. URL: https://doi.org/10.1145/3267305.3267689.

[14] Ming Jin et al. "PresenceSense: Zero-Training Algorithm for Individual Presence Detection Based on Power Monitoring". In: *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings.* BuildSys '14. Memphis, Tennessee: Association for Computing Machinery, 2014, 1–10. ISBN: 9781450331449. DOI: 10.1145/2674061.2674073. URL: https://doi.org/10.1145/2674061.2674073.

[15] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization.* 2017. arXiv: 1412.6980 [cs.LG].

[16] Steven R. Livingstone and Frank A. Russo. “The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English”. In: *PLOS ONE* 13.5 (May 2018), pp. 1–35. DOI: 10.1371/journal.pone.0196391. URL: https://doi.org/10.1371/journal.pone.0196391.

[17] Niluthpol Chowdhury Mithun, Sujoy Paul, and Amit K. Roy-Chowdhury. “Weakly Supervised Video Moment Retrieval From Text Queries”. In: *CoRR* abs/1904.03282 (2019). arXiv: 1904.03282. URL: http://arxiv.org/abs/1904.03282.

[18] Mustaqeem and Soonil Kwon. “A CNN-Assisted Enhanced Audio Signal Processing for Speech Emotion Recognition”. In: *Sensors* 20.1 (2020). ISSN: 1424-8220. DOI: 10.3390/s20010183. URL: https://www.mdpi.com/1424-8220/20/1/183.

[19] V.V. Nanavare and S.K. Jagtap. “Recognition of Human Emotions from Speech Processing”. In: *Procedia Computer Science* 49 (2015). Proceedings of 4th International Conference on Advances in Computing, Communication and Control (ICAC3’15), pp. 24–32. ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2015.04.223. URL: https://www.sciencedirect.com/science/article/pii/S1877050915007322.

[20] *Neural Networks for Machine Learning, class slides.*

[21] Sandeep Kumar Pandey, Hanumant Singh Shekhawat, and S. R. Mahadeva Prasanna. “Deep Learning Techniques for Speech Emotion Recognition: A Review”. In: *2019 29th International Conference Radioelektronika (RADIOELEKTRONIKA)* (2019), pp. 1–6.

[22] Sujoy Paul, Sourya Roy, and Amit K. Roy-Chowdhury. “W-TALC: Weakly-supervised Temporal Activity Localization and Classification”. In: *CoRR* abs/1807.10418 (2018). arXiv: 1807.10418. URL: http://arxiv.org/abs/1807.10418.

[23] M. Kathleen Pichora-Fuller and Kate Dupuis. *Toronto emotional speech set (TESS)*. Version DRAFT VERSION. 2020. DOI: 10.5683/SP2/E8H2MF. URL: https://doi.org/10.5683/SP2/E8H2MF.

[24] Yuji Roh, Geon Heo, and Steven Euijong Whang. “A Survey on Data Collection for Machine Learning: a Big Data - AI Integration Perspective”. In: *CoRR* abs/1811.03402 (2018). arXiv: 1811.03402. URL: http://arxiv.org/abs/1811.03402.

[25] Ervin Sejdić, Igor Djurović, and Jin Jiang. "Time–frequency feature representation using energy concentration: An overview of recent advances". In: *Digital Signal Processing* 19.1 (2009), pp. 153–183. ISSN: 1051-2004. DOI: https://doi.org/10.1016/j.dsp.2007.12.004. URL: https://www.sciencedirect.com/science/article/pii/S105120040800002X.

[26] Connor Shorten and Taghi M. Khoshgoftaar. "A survey on Image Data Augmentation for Deep Learning". In: *Journal of Big Data* 6.1 (2019), p. 60. ISSN: 2196-1115. DOI: 10.1186/s40537-019-0197-0. URL: https://doi.org/10.1186/s40537-019-0197-0.

[27] Kevin Tomba. et al. "Stress Detection Through Speech Analysis". In: *Proceedings of the 15th International Joint Conference on e-Business and Telecommunications - ICETE,* INSTICC. SciTePress, 2018, pp. 394–398. ISBN: 978-989-758-319-3. DOI: 10.5220/0006855805600564.

[28] George Trigeorgis et al. "Adieu features? End-to-end speech emotion recognition using a deep convolutional recurrent network". In: *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 2016, pp. 5200–5204. DOI: 10.1109/ICASSP.2016.7472669.

[29] M.V. Valueva et al. "Application of the residue number system to reduce hardware costs of the convolutional neural network implementation". In: *Mathematics and Computers in Simulation* 177 (2020), pp. 232–243. ISSN: 0378-4754. DOI: https://doi.org/10.1016/j.matcom.2020.04.031. URL: https://www.sciencedirect.com/science/article/pii/S0378475420301580.

[30] Yongjin Wang and Ling Guan. "Recognizing human emotion from audiovisual information". In: *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.* Vol. 2. IEEE. 2005, pp. ii–1125.

[31] Wikipedia contributors. *Mel scale — Wikipedia, The Free Encyclopedia.* 2021. URL: https://en.wikipedia.org/w/index.php?title=Mel_scale&oldid=1061063889.

[32] Wikipedia contributors. *Stochastic gradient descent — Wikipedia, The Free Encyclopedia.* 2022. URL: https://en.wikipedia.org/w/index.php?title=Stochastic_gradient_descent&oldid=1064704823.

[33] Wikipedia contributors. *Weak supervision — Wikipedia, The Free Encyclopedia.* 2022. URL: https://en.wikipedia.org/w/index.php?title=Weak_supervision&oldid=1071121496.

[34] Wikipedia contributors. *Zero-crossing rate — Wikipedia, The Free Encyclopedia.* 2021. URL: `https://en.wikipedia.org/w/index.php?title=Zero-crossing_rate&oldid=1020703960`.

[35] Biqiao Zhang, Emily Mower Provost, and Georg Essl. "Cross-corpus acoustic emotion recognition from singing and speaking: A multi-task learning approach". In: *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 2016, pp. 5805–5809. DOI: `10.1109/ICASSP.2016.7472790`.

[36] Wei Zhang et al. "Parallel distributed processing model with local space-invariant interconnections and its optical architecture". In: *Appl. Opt.* 29.32 (1990), pp. 4790–4797. DOI: `10.1364/AO.29.004790`. URL: `http://opg.optica.org/ao/abstract.cfm?URI=ao-29-32-4790`.

[37] Jianfeng Zhao, Xia Mao, and Lijiang Chen. "Speech emotion recognition using deep 1D & 2D CNN LSTM networks". In: *Biomed. Signal Process. Control.* 47 (2019), pp. 312–323.