



University of Glasgow | School of  
Computing Science

# **Deep learning for robust dimensional characterisation of affect in speech**

Wesley Scott

School of Computing Science  
Sir Alwyn Williams Building  
University of Glasgow  
G12 8QQ

A dissertation presented in part fulfilment of the requirements of the  
Degree of Master of Science at The University of Glasgow

21 September 2020

## **Abstract**

This paper seeks to evaluate the usage of machine learning methodology in the task of speech emotion recognition (SER) by utilising a signal processing approach. This is done with aim of assessing user behaviour exhibited during online voice communication.

We use dimensional models of emotion, put forward by empirical research, to gain nuanced information about sampled emotion data, and to disseminate greater insight into user voice communication activity.


This paper focuses on the extraction and use of Mel-frequency cepstral coefficients (MFCC) as feature vectors for feed forward neural network architectures.

The conducted experiments show evidence that the methodology proposed in this paper is partially effective on unseen, dissimilar in structure real-world data, which has proven to be a hurdle to deployment of solutions in the area of automatic speech recognition (ASR) and SER. These findings provide a framework to enable more precise, automated user handling.

## Education Use Consent

I hereby give my permission for this project to be shown to other University of Glasgow students and to be distributed in an electronic format. **Please note that you are under no obligation to sign this declaration, but doing so would help future students.**

Name: Wesley Scott

Signature: 

## Acknowledgements

I would like to thank Dr. John H. Williamson at the University of Glasgow for supervising my research under the difficult circumstances of the COVID-19 pandemic, as well as invaluable insight as to how to realise and build upon the initial concept of this paper.

Thanks go to Dr. Don Knox for his instruction during completion of the Audio Analysis module as part of the Audio Technology undergraduate programme at Glasgow Caledonian University, as well as assistance with enrollment to the MSc conversion programme at the University of Glasgow.

# Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
1.1	Concept . . . . .	4
1.2	Objectives . . . . .	4
<b>2</b>	<b>Analysis &amp; Approach</b>	<b>5</b>
2.1	Background . . . . .	5
2.1.1	Voice Communication and Online Gaming . . . . .	5
2.1.2	Current Solutions . . . . .	5
2.1.3	Emotion as a behavioural measure . . . . .	6
2.2	Requirements . . . . .	7
2.2.1	Machine learning approach . . . . .	7
2.2.2	Pre-processing . . . . .	8
2.2.3	Feature Selection and Extraction . . . . .	10
2.2.4	Utilised data-sets . . . . .	11
2.2.5	Data-set Augmentation . . . . .	12
2.2.6	Dimensional Modelling . . . . .	15
2.3	Aims . . . . .	16
<b>3</b>	<b>Design Implementation</b>	<b>17</b>
3.1	Implementation Details . . . . .	17
3.1.1	Data handling . . . . .	17
3.1.2	Neural Network Architecture . . . . .	19

<b>4</b>	<b>Testing &amp; Evaluation</b>	<b>20</b>
4.1	Strategy . . . . .	20
4.2	Results . . . . .	20
4.2.1	Error score . . . . .	20
4.2.2	Discussion . . . . .	20
<b>5</b>	<b>Conclusions</b>	<b>24</b>
5.1	Summary of Results . . . . .	25
5.2	Further Work . . . . .	25
5.2.1	Deployment . . . . .	25
5.2.2	Current deep learning methodology . . . . .	25

# Chapter 1

## Introduction

This introductory chapter discusses the reasoning behind the project’s concept, with regards to difficulty faced in ensuring universally positive user experiences within online voice communication. Additionally, it outlines how experiences can be related to emotion vocalisation. Finally, it describes the proposed methods for recognising emotion cues.

### 1.1 Concept

The developing popularity of competitive video games (‘Esports’) and their reliance on voice communication in their core game-play loops has led to compromise regarding the social content of those communications.

In this virtual setting, moderation of user behaviour relies heavily on manual user feedback to guide social metrics. If a user is subject to an undesirable experience specifically as the result of using packaged in-game voice communications, the standard solution in Esports titles to mitigate such disturbance is to manually mute communications on the client side from the offending sources. This often has the adverse effect of any useful communications those sources may provide being lost in the process, leading to an overall limited experience of the game for users during that session. Most competitive games implement a system whereby all users are provided with the ability to create a report ticket regarding another user. This can be done at any time during, or post-game session; users can report if other users are exhibiting negative or undesirable vocal outbursts.

### 1.2 Objectives

This paper proposes a more overarching effort to harness user behavioural data and to identify vocalised emotion in order to score voice communication activity, which in turn can be used to create a bespoke solution in addition to other forms of user handling.

The implications of such solution achieving success are relevant to various other areas where SER may prove invaluable, such as recommender systems and customer service applications. In these

areas, manual user data entry usually provides a major source of feedback for system implementation. By incorporating automated user emotion recognition, more nuanced user studies can be carried out. Alongside other user review metrics, such studies could be used to assess the presence of negative detractors to global user experience.

## **Chapter 2**

# **Analysis & Approach**

### **2.1 Background**

#### **2.1.1 Voice Communication and Online Gaming**

Attempting to define what is acceptable behaviour in voice communication during online gaming shows that there exists a normalisation of negative behaviour in this context, with seventy-four percent of adults who play online multiplayer games in the US experiencing some form of not only negative user experience, but some form of harassment during game-play too. In the conducted survey only 27 percent of online multiplayer game users reported that harassment had not impacted their game experience at all, 73 percent of players having had their experience shaped by harassment in some way [Lea19]. Though a user may have a neutral or even positive reputation through existing review metrics, this alone cannot for certain determine a users continued responsible use of voice communication privileges, nor account for edge case anomalies in which generally trusted users abuse voice communication channels. In order to account for this, additional measures specific to voice communication are valuable to those interested in user experience (UX) design considerations.

#### **2.1.2 Current Solutions**

There are efforts to prevent inappropriate use of voice communication from overtly disturbing the balance of user experience whilst gaming. These rely on user created reports, where a significant number of reports levied against a single user results in a suspension of their voice communication privileges [Sof20]. Though this approach has been demonstrated to be partially effective, it can be argued that being matched with a user without voice communication privileges adversely affects other users within a team setting. It is often desirable that crucial game-play information is conversed, albeit that it may come with some negative input. This is the current status-quo in many popular Esports titles. The aim of this paper is to potentially provide UX designers with a system

based on dimensional modelling with which undesirable behaviour can be more precisely measured and mitigated.



Figure 2.1: User report form, Counter-Strike: Global Offensive [Jim]

In *Exploring Cyberbullying and Other Toxic Behavior in Team Competition Online Games*, Kwak et al. discuss the main issues that cause manual user feedback to be ineffective in tackling disruptive behaviour. It has been found that there are biases with respect to reporting teammates vs. opponents and that users are generally not engaged in actively reporting poor usage of communication channels. "The low degree of participation for social control we found raises a fundamental question about the design considerations of such report-based systems: if relatively few "victims" voluntarily report such behavior, how effective can it truly be?" [KBH15]. The paper also discusses the issue of false reports which lead to innocent users being reprimanded for assumed poor behaviour. This is somewhat mitigated by crowd-sourced tribunal systems implemented in some Esports titles, whereby other players review reports to confirm authenticity. In *Predicting crowd-sourced decisions on toxic behavior in online games*, Blackburn et al. conclude that it is clear that crowd-sourcing using experienced users is useful in protecting innocent victims who are wrongly reported by other users. The tribunal system could be further improved in regard to quality and efficiency via several proposed mechanisms for quality control in crowd-sourced systems and augmentation with machine-learning solutions [BK14]. Our study would propose such a solution, which is to be used in conjunction with existing user handling mechanisms.

### 2.1.3 Emotion as a behavioural measure

In *Negative emotions and behaviour: The role of regulatory emotional self-efficacy* [MVM18], Mesurado et al. analysed the direct effect of negative emotions (anger, depression and anxiety) on pro-social and aggressive behaviour, which expresses the affect link between emotion and behaviour required for progressing SER for our task. The results of the study show that anger has a direct relationship with pro-social behaviour and aggression. Similar studies have been conducted



with regard to high-stress situations. Negative Emotions and Their Effect on Customer Complaint Behaviour identifies the kinds of emotion linked to a particular negative experience, and the kinds of behaviour likely to stem from such an experience [Tro11]. These situational stimuli are akin to social situations we might observe in competitive gaming.

## **2.2 Requirements**

### **2.2.1 Machine learning approach**

Optimisations in compute performance have ushered a new era in the field of machine learning and particularly in deep learning, which in turn has contributed to advancement in many areas of research, such as computer vision and audio analysis. Before deep learning, ASR problems were handled using techniques such as Hidden Markov Modelling and Gaussian Mixture Modelling. ASR has been an area with strides being made on the back of deep learning and commercial speech recognition implementations for automation, such as Amazon's Alexa and professional dictation applications.

The drive of this advancement comes from the ability to train deep neural network architectures via powerful central processing unit (CPU) and graphics processing unit (GPU) hardware acceleration. Neural networks are a topical area of machine learning currently, that focus on creating layered models with node constructs at each layer. Layers between the input and the output of a network are referred to as 'hidden layers'.

The term node in the context of neural networks refers to a representation of a mathematical function, which is multiplied by values called 'weights' and summed with a learnable parameter 'bias'. Nodes with similar purpose are grouped into layers. A node which belongs to a given layer will pass its output as input to the nodes it is connected to from the next hidden layer, or simply output the result value if it belongs to the final output layer.

A non-linear function, also known as an activation function, is applied to generate a given nodes output. The training process consists of gradually updating the weights and biases of a network so that a function of the network's output, known as the loss function, is minimised. The gradient of the loss function with respect to the weights of the network is computed using back propagation [RHW86], it is then used within an optimiser, traditionally Stochastic Gradient Descent, or more recently, with an adaptive gradient descent approach such as Adam [KB14].

#### **2.2.1.1 Generalisation error & over-fit**

Large neural networks trained on relatively small data-sets can 'over-fit' on the training data. This has the effect of the model learning the statistical noise in the training data, which can result in the loss function being less predictive of the true task when the model is evaluated on unseen data.

### **2.2.1.2 Batch Normalisation**

Batch normalisation, proposed by Ioffe and Szegedy [IS15], mimics the way data is often normalised prior to use in machine learning, to provide a single distribution. It is used when training deep neural networks to help standardise the inputs when using batch methodology for data loading. This can have the effect of stabilising the learning process, with potential to reduce the number of training iterations, generally referred to as epochs, required to train a neural network and enabling the use of heightened learning rates.

There are two main methods in which batch normalisation is applied within a forward function, either before the activation function (non-linearity) or after it, following the relu activation for example. In this paper we follow the implementation guidance of the authors of the original paper [IS15] and apply batch normalisation immediately after the linear layers.

We must also consider how to parameterise our batch normalisation with respect to momentum. Momentum is the importance given to the previous moving average, when calculating the population average for inference. For smaller batch sizes, a high momentum value is generally used. The number of steps per epoch will be higher, so high momentum results in a slow, but steady learning of the moving mean. As there will be less steps per epoch with larger batch sizes, the same does not hold true. The statistics in that case are more similar as that of the entire data set. At these times, momentum has to be less, so that the mean and variance are updated more quickly.

### **2.2.1.3 Dropout**

Dropout as a documented technique for training neural networks was coined in [SHK<sup>+</sup>14] and is another technique used to improve generalisation of machine learning models and avoid overfit. Dropout attempts to allow a single model architecture to be used to simulate having a large number of different network architectures by randomly dropping out neural nodes during training. Probabilistic use of dropout has been shown as a simple and effective regularisation method for larger networks [SHK<sup>+</sup>14].

### **2.2.1.4 Combining Generalisation Techniques**

Researchers discuss the issues with using batch normalisation and dropout methodologies in conjunction with one another [LCHY19], with the conclusion drawn that dropout should be used only after all the batch normalisation layers. This finding is later analysed in our evaluation of neural network architectures for our SER task.

## **2.2.2 Pre-processing**

The aim of machine learning is to extract features from data and create a dense representation of the content. This forces the machine learning model to learn the core information without noise in order to make inferences on a given set of data.

Before extraction of features there are some useful steps which assist in separating noise in this sense for training a machine learning model for SER.

#### 2.2.2.1 Pre-emphasis

Our first transform is a pre-emphasis filter that amplifies higher frequencies in our signal, given as

$$y_t = x_t - \alpha x_{t-1} \quad (2.1)$$

where the denominator  $\alpha = 0.97$

This process is performed to compensate for the average magnitude of a given audio signal, by emphasizing mid and high frequencies to the loss function of a machine learning model. Studies suggest that neural networks can struggle at modeling high-frequency content introduced by distortion effects without a pre-emphasis stage [DJV<sup>+</sup>19].

#### 2.2.2.2 Silence removal/threshold

We also apply silence removal, whereby we discard the less useful parts of our audio samples. We could use the silence to try to model certain speech patterns, but often the amount and length of silence in speech depends highly on context. We want our model to identify an emotional vocalisation independent of the context in which the voice sample was taken (i.e. shouting, whispering, fast, slow). This is more difficult to train, but should make the machine learning model more robust to prediction of real-world data.

In this case we are interested in parts of an audio clip above the absolute value of 30dB as our threshold, based on typical recording situations from the available data-sets.

#### 2.2.2.3 Framing with Fast Fourier Transform

The frequencies of a given speech audio signal change over time. To account for this, we look at the frequencies of our signal at different frames or windows across the signal. This is done with the aim of preserving the frequency contours across the signal through time.

This forces us to make the assumption that frequencies in a signal are stationary over a short time frame. Under this assumption, by doing an FFT (Fast Fourier Transform) over this short-time frame, we can obtain a useful approximate of the frequency contours of the signal by concatenating adjacent frames.

An FFT is measurement method converting a signal into individual spectral components and providing frequency information about the signal. We sample our audio at 16kHz, which provides an audible spectrum of 8kHz as given by Nyquist theorem. This provides ample bandwidth to extract information as the crucial information in speech change appears at lower frequency bands [NMS<sup>+</sup>18], [DAKD12].

Having split a given original signal into short frames, we apply a window function to each frame. This functions to counteract the FFT assumption that the data is infinite, provides an accurate representation of the pre-windowed signal's frequency spectrum, and reduces potential spectral leakage. The Hamming window is used extensively in speech processing application, providing a balance between frequency resolution and dynamic range. The Hamming window is given as

$$w[n] = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right) \quad (2.2)$$

where  $0 \leq n \leq N-1$ ,  $N$  is the window length.

#### 2.2.2.4 Short-Time Fourier Transform (STFT)

We perform an  $N$ -point FFT on each frame to calculate the frequency spectrum. Calculating FFT on sliding frames is referred to as Short-Time Fourier Transform (STFT). We then compute the power spectrum given by

$$P = \frac{|FFT(x_i)|^2}{N} \quad (2.3)$$

where  $x_i$  is the  $i^{th}$  frame of the signal  $x$ .

#### 2.2.2.5 Mel-scale

We apply a triangular filter on a Mel-scale to the power spectrum in order to extract frequency bands. The Mel-Scale models the non-linear way in which the human ear perceives sound. The human ear is more discriminate at lower frequencies and less discriminate at higher frequencies based on the frequency response of the cochlea. We convert between frequency in Hertz ( $f$ ) and Mel ( $m$ ) as follows

$$m = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \quad (2.4)$$

$$f = 700(10^{m/2595} - 1) \quad (2.5)$$

### 2.2.3 Feature Selection and Extraction

#### 2.2.3.1 Mel-Frequency Cepstral Coefficients (MFCC)

Mel filter bank coefficients are highly correlated, which can lead to issues during training with machine learning models. To provide our machine learning model with a usable feature set with which to associate an emotion, Mel-Frequency Cepstral Coefficients (MFCC) are a suitable candidate, widely used in the area of ASR [HCCP06]. We apply a Discrete Cosine Transform (DCT) to de-correlate the filter bank coefficients and provide a compressed representation of the filter banks.

These coefficients are more robust and reliable to variations speakers and recording conditions [Dav13].

We derive the MFCC for our audio clips by time series, extracting 39 from each clip to allow static input dimensions for our machine learning model. Traditional MFCC feature sets use only 12 cepstral coefficients (F1-13). The zeroth coefficient (F0) is often excluded as it represents the average log-energy of the input signal, which carries less useful speaker-specific information. Additional information regarding the temporal dynamics of the signal is obtained by computing the first and second derivatives of the cepstral coefficients. The first-order derivative are the delta (differential), and the second-order derivative are the delta-delta (acceleration) coefficients. These values can provide our machine learning model information about speech rate. The delta coefficients are given by

$$d_t = \frac{\sum_{n=1}^N n (c_{t+n} - c_{t-n})}{2 \sum_{n=1}^N n^2} \quad (2.6)$$

where  $d_t$  is a delta coefficient from a frame  $t$  computed in by the static coefficients  $c_{t-n}$  to  $c_{t+n}$ . The acceleration coefficients are computed similarly, but using the differential instead of the static coefficients. We take F1 through F40 of each window of an audio clip to construct our feature set.

### 2.2.3.2 Feature labelling

The fact that emotions are dynamic in nature and evolve across time has been explored relatively less often in automatic SER systems to date [HE18]. Understanding the nature of emotional change in speech is crucial to effective feature extraction for our machine learning model, and should be a consideration when deciding the length and frequency of our FFT windows. Typically speech sequences are pre-segmented into small utterances with one global category or dimension label for each. However, this per-utterance labeling is based on the assumption that emotions are in steady-state across the whole utterance, while emotions are dynamic in nature and change over time [Sch05]. Typically very short windowing and strides (time in between beginning of next window) are used in ASR, corresponding to the measure of phones in speech. For SER tasks, a slightly different approach is required. In this paper we use the window size of 0.5 seconds and a stride (how often a new window begins) of 0.1 seconds. Through initial experiments a shorter window length proved ineffective for training, and this was found to be optimal through fine-tuning parameters.

### 2.2.4 Utilised data-sets

In order to train a machine learning model for SER, several data-sets collated for similar studies were selected for training, validation and testing purposes.

The main difficulty with ASR models is applying solutions to the task of predicting real-world data as opposed to similarly structured control data to that of the model’s training set. In order to obtain useful outputs from our model with accurate test scores on wild data, training and validation data is selected with a number of variables, particularly focused on including a number of speakers, with varying recording conditions.

#### **2.2.4.1 RAVDESS**

The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) [LR18] of which only the speech samples are utilised in the data for this paper. Speech samples includes calm, happy, sad, angry, fearful, surprise, and disgust expressions, performed by 12 male and 12 female professional actors. The aim of including RAVDESS in our train and validation set is to have a clear definition of each emotion via professional vocalisation. RAVDESS also includes more discrete emotional classes than the other data-sets, introducing 'calm' as a nuance to a neutral state. All 1440 original clips are used within our train, validation and test split.

#### **2.2.4.2 CREMA-D**

Crowd-Sourced Emotional Multimodal Actors data-set (CREMA-D) [CCK<sup>+</sup>14] provides the most diverse number of speakers, 91 actors (48 male and 43 female), aged between 20 and 74, from a variety of races and ethnicities, African America, Asian, Caucasian, Hispanic, and Unspecified. CREMA-D makes up the majority of the train, validation and test split, with 7,442 original clips.

#### **2.2.4.3 TESS**

Toronto emotional speech set (TESS) [PFD20] stimuli were modeled on the Northwestern University Auditory Test No. 6 [TC66]. A set of 200 target words were spoken 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.

Data-sets such as TESS, prove less useful to work with in isolation for training a model upon initial experiments for our purposes, as the number of recorded speakers is limited; moreover, the intonation of the spoken lines, such as with TESS carrier phrase method "Say the word .....", is repetitive. When working with audio features such patterns are easily learned and lead to over-fit on the training data, therefore resulting in poor prediction of real-world online voice communication data.

Given these considerations, we take the most varied data-sets, CREMA-D and RAVDESS as our model training and validation data. We then use TESS as emulation of a wild test set, where emotional cues are clearly defined, and our model is not pre-disposed to the structured nature or pattern of the speech samples.

#### **2.2.5 Data-set Augmentation**

Augmentation methods are used to generate additional samples from our chosen training and validation data-sets to improve generalisation. Methods were chosen both to prevent our machine learning model from over-fitting as well as to emulate similar real-world application scenarios. As it is common that users may not manually calibrate their microphones for the best performance of

other users, we augment to best represent real-world signal scenarios. Silence is also removed in batch from all training samples where applicable, in order to avoid unnecessary feature extraction which may harm results [KRB08].

A poor signal-to-noise ratio (SNR) can arise with incorrect gain values which results in compensation using compression. The compression lowers amplitude of the offending frequencies over the given threshold, but in turn increases the chance of audible signal noise. A SNR of 20:1 was used during mixing of augmentation techniques and root mean squared (RMS) calculated through each signal to ensure uniform application of noise throughout our data.

### 2.2.5.1 Additive white Gaussian noise (AWGN)

Gaussian white noise provides a noise source of equal intensity over all sampled frequencies, helpful in avoiding over-fitting while training a machine learning model. A pure white noise source is unlikely to appear in a real-world recording scenario, however provides a unique and identifiable transform for our machine learning model.

The following formula provides the required AWGN that should be added to the signal to achieve desired SNR (20:1).

First, calculate the RMS of the signal

$$RMS_{signal} = \sqrt{\text{mean}(signal^2)} \quad (2.7)$$

Based on these results, calculate the required RMS of noise signal to be generated

$$RMS_{noise} = \sqrt{\frac{RMS_{signal}^2}{\frac{SNR^{10}}{20}}} \quad (2.8)$$

As we are using AWGN, mean is equal to 0, therefore STD is equal to RMS

$$STD_{noise} = RMS_{noise} \quad (2.9)$$

We then generate the noise signal, to be returned a combined with original input signal.

The probability density for the Gaussian distribution:

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (2.10)$$

where  $\mu$  is 0, and  $\sigma$  is the standard deviation of the noise ( $STD_{noise}$ ).

### 2.2.5.2 Simulated chatter, miscellaneous background noise

A custom background noise set was created to emulate off-axis background speakers, typical of a real-world voice communication setting without isolation of the speaker, noise cancellation or where a noise gating option is not available or threshold not sufficiently calibrated via the user.

The following steps provide the required background noise that should be added to the signal to achieve desired SNR (20:1). First, calculate the RMS of the signal (seen in equation 2.1). Based on these results, calculate the required RMS of noise signal to be generated (seen in equation 2.2).

We then require measurement of the current RMS of the noise source

$$RMS_{currentnoise} = \sqrt{\text{mean}(noise^2)} \quad (2.11)$$

Finally, generate the noise signal, to be returned and combined with original input signal

$$noise = noise \frac{RMS_{noise}}{RMS_{currentnoise}} \quad (2.12)$$

### 2.2.5.3 Reverberation, simulated room modality

Algorithmically applied copies of the data-sets have been generated to simulate room modality where poor calibration has been performed to separate external noise from entering the microphone input source. In this case, we try to emulate a situation where a small to medium sized room is used and reflections are output to voice communication channels.

In order to calculate appropriate room impulse responses for augmentation we use Sabine, where RT60 is the time which the signal takes to decay -60dB. This establishes a relationship between the T60 of a room, its volume, and its total absorption (in sabins). This is given by the equation:

$$T_{60} = \frac{24 \ln 10^1}{c_{20}} \frac{V}{Sa} \approx 0.1611 \text{ sm}^{-1} \frac{V}{Sa} \quad (2.13)$$

$c_{20}$  is the speed of sound in the room, V is the volume of the room in meters cubed, S total surface area of room in meters squared, a is the average absorption coefficient of room surfaces, and the product Sa is the total absorption in sabins.



#### 2.2.5.4 Overdrive/Distortion (signal clipping and colouration)

A typical issue with online voice communications is that a lack of correct client side gain, compression, limiting or normalisation setting can lead to clipping whereby there is no 'headroom' for the signal. In this case clipping leads to distortion in the signal, therefore we model this effect to simulate these conditions.

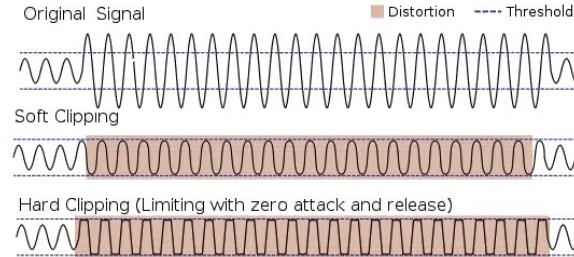


Figure 2.2: Examples of signal clipping in the time domain [Mik]

#### 2.2.6 Dimensional Modelling

A comprehensive collection of research in the area of SER has been compiled in *Survey on speech emotion recognition: Features, classification schemes, and databases* [EAKK11]. Ayadi et. al. discuss the difficulty in deciding a corpora of emotion for emotion recognition tasks. Sets such as Schubiger [Sch58] and O'Connor and Arnold [AO73] number 300 in discrete emotional states, and classifying such a large number of emotions has proved very difficult in practise [EAKK11]. Archetypal emotions, emotions which are considered distinct in everyday life, are often chosen to reduce output dimensions, this is a practise generally agreed upon by researchers. Often 'palette theory' is employed, which works on the assumption that any emotion can be decomposed into primary emotions, these are *anger*, *disgust*, *fear*, *joy*, *sadness*, and *surprise* [CDCT<sup>+</sup>01].

As it is desired to use a dimensional model of emotion rather than discrete classes to gain nuanced information about speech samples and their content, available data-sets which are labelled using a form of discrete emotion theory must be manually mapped according to the desired dimensional modelling methodology [Lan18]. To achieve this, figures from Russel's circumplex model paper [Rus80] are used to map the available emotion classes to X, Y coordinates, which provide the ground truth for our machine learning model.

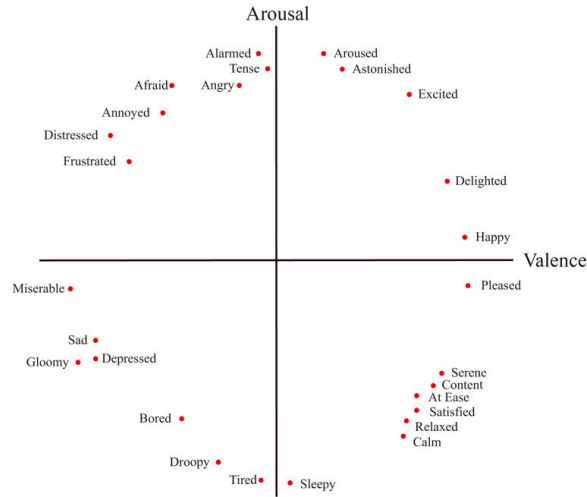


Figure 2.3: Russel's Circumplex Model [Rus80]

As not every emotion of the circumplex model is accounted for in the available data-sets, we map what is available to us to provide context. When predicting, we make assumptions about a given sample based on its predicted X, Y coordinate in the dimensional space [Lan18]. These X, Y coordinates relate to the axis of valence on the X axis and arousal on the Y axis.

## 2.3 Aims

Given the proposed methods in our requirements, the following are the desired outcomes of this paper:

- Train a deep learning model capable of predicting emotion from speech recording data
- Map a given prediction to coordinates on a dimensional model of emotion.
- Use train, validation, test and emulated wild test data to assess the model's usefulness in the application of recognising emotion in real-world voice communication audio data.

## Chapter 3

# Design Implementation

### 3.1 Implementation Details

#### 3.1.1 Data handling

To provide our features, we pre-process and derive the MFCC for our audio clips, extracting exactly 39 from each clip to allow for static input dimension for our machine learning model. Alongside this, for each FFT window of 39 features, an emotion label ground truth. This process is performed on clean and augmented sets. We make use of stratified random seed sampling to ensure a repeatable splitting of this data, as well as a distribution of samples for our training, validation and test splits which preserves the same proportions of examples in each class as observed in the original data-sets.

In total 44410 audio clips are used for the initial train, validation and test splits, made up of the following: RAVDESS clean, CREMA-D clean plus four sets of augmented files for RAVDESS and CREMA-D. These files are split using the following ratios: 62.5 percent training, 25 percent validation and 2.5 percent kept for a test plot. 10 percent of the overall data is dropped in stratified fashion to reduce the number of eventual samples provided by our chosen FFT window size.

We also utilise our emulated wild test set, made up of 20 percent of the samples from the TESS data-set. This provides insight into performance on data which does not share the same structure or speakers as the initial training and validation sets. The breakdown of our data use is as follows:

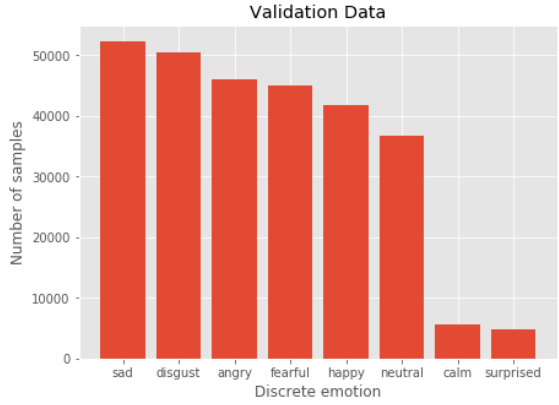
Table 3.1: Table of data utilisation

Data Category	Count
Original audio clips	44410
Features per sample	39
Samples provided via FFT	829400
Training samples	518375
Validation samples	282750
Test samples	28275
'Wild' samples	7533



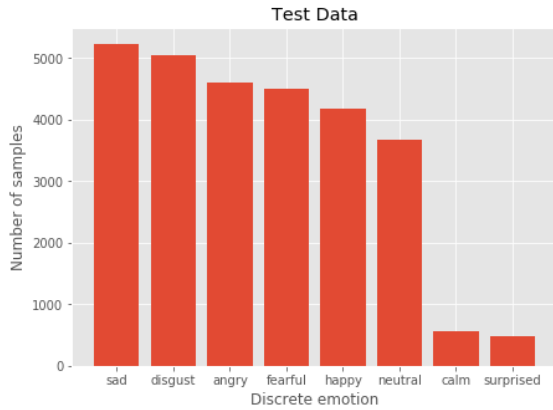
(a) Table of train distribution

Discrete Emotion	Sample count
Sad	95906
Disgust	92638
Angry	84251
Fearful	82581
Happy	76653
Neutral	67300
Calm	10123
Surprised	8923



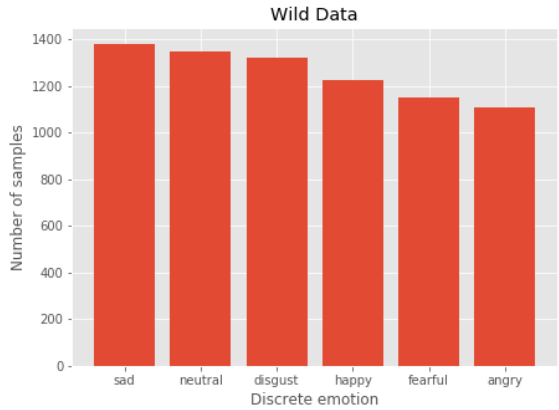
(b) Table of validation distribution

Discrete Emotion	Sample count
Sad	52313
Disgust	50530
Angry	45955
Fearful	45045
Happy	41810
Neutral	36709
Calm	5522
Surprised	4866



(c) Table of test distribution

Discrete Emotion	Sample count
Sad	5231
Disgust	5053
Angry	4596
Fearful	4504
Happy	4181
Neutral	3671
Calm	552
Surprised	487



(d) Table of 'wild' distribution

Discrete Emotion	Sample count
Sad	1379
Disgust	1322
Angry	1107
Fearful	1152
Happy	1226
Neutral	1347
Calm	Not included in data-set
Surprised	Not included in data-set

Figure 3.1: Train, validation and test split of utilised data-sets

A larger than normal validation sample size is chosen due to the large number of samples per original discrete emotional class which is over 5000 clips per class in some cases, providing a better evaluation based on the batch size of 256 samples.

### **3.1.2 Neural Network Architecture**

A simple multi-layer perceptron (MLP) model is selected to determine the level of recognition possible from a basic neural network structure, utilising the PyTorch framework. With this multi-layered model we can evaluate useful optimisation and generalisation techniques which may prove beneficial for training neural networks in the task of emotion recognition with speech data. In order to optimise our models we use a means squared error (MSE) loss function provided by the PyTorch framework, optimising via Adam [KB14].

#### **3.1.2.1 Parameter Optimisation**

In order to achieve optimal results from our network using our optimiser, we perform best estimator grid search queries in order to find the optimal size for our hidden layers prior to further evaluation. We use a learning rate of 0.001, which is based upon the best estimator grid search.

#### **3.1.2.2 Baseline PyTorch MLP**

Within our networks sequential forward function is first a linear layer, taking the input dimensions of our features (39), and the output dimension of our first hidden layer provided by best estimator search results (300). This is followed by a non-linear relu activation, for input to the second linear layer, which has input dimension 300, and output dimensions of our second set of hidden layers (150). This is again followed by a non-linear relu activation, and finally passed to a linear output, taking input dimensions of our second set of hidden layers (150) and outputting to our required coordinate dimension of 2 to provide continuous X, Y outputs.

#### **3.1.2.3 Utilising Batch Normalisation**

To improve our initial model, we build upon the sequential forward function by adding batch normalisation functions after the first two linear layers in the network, before the relu activations. We work with a conservative value of momentum 0.6 after fine tuning experimentation, likely due to the slightly larger batch size use of 256 provided by our best estimator grid search.

#### **3.1.2.4 Utilising Dropout**

Finally, in order to force our model to generalise with aim of providing more inference in its prediction results, we implement dropout into the network. One dropout layer is added immediately

before the final linear output layer, as discussed in [LCHY19], with a conservative probability ( $p$ ) value of 0.175.

## Chapter 4

# Testing & Evaluation

### 4.1 Strategy

In order to test and evaluate effectively our various model prediction performances to the corresponding circumplex dimensional model coordinates, we provide the mean squared error (MSE) score. Based on the convergence of training and validation loss timings of our baseline model, we train for 100 epochs and take the scores of predictions made on train, validation, test and emulated wild test data.

### 4.2 Results

#### 4.2.1 Error score

Table 4.1: Table of results comparing neural network configurations (MSE)

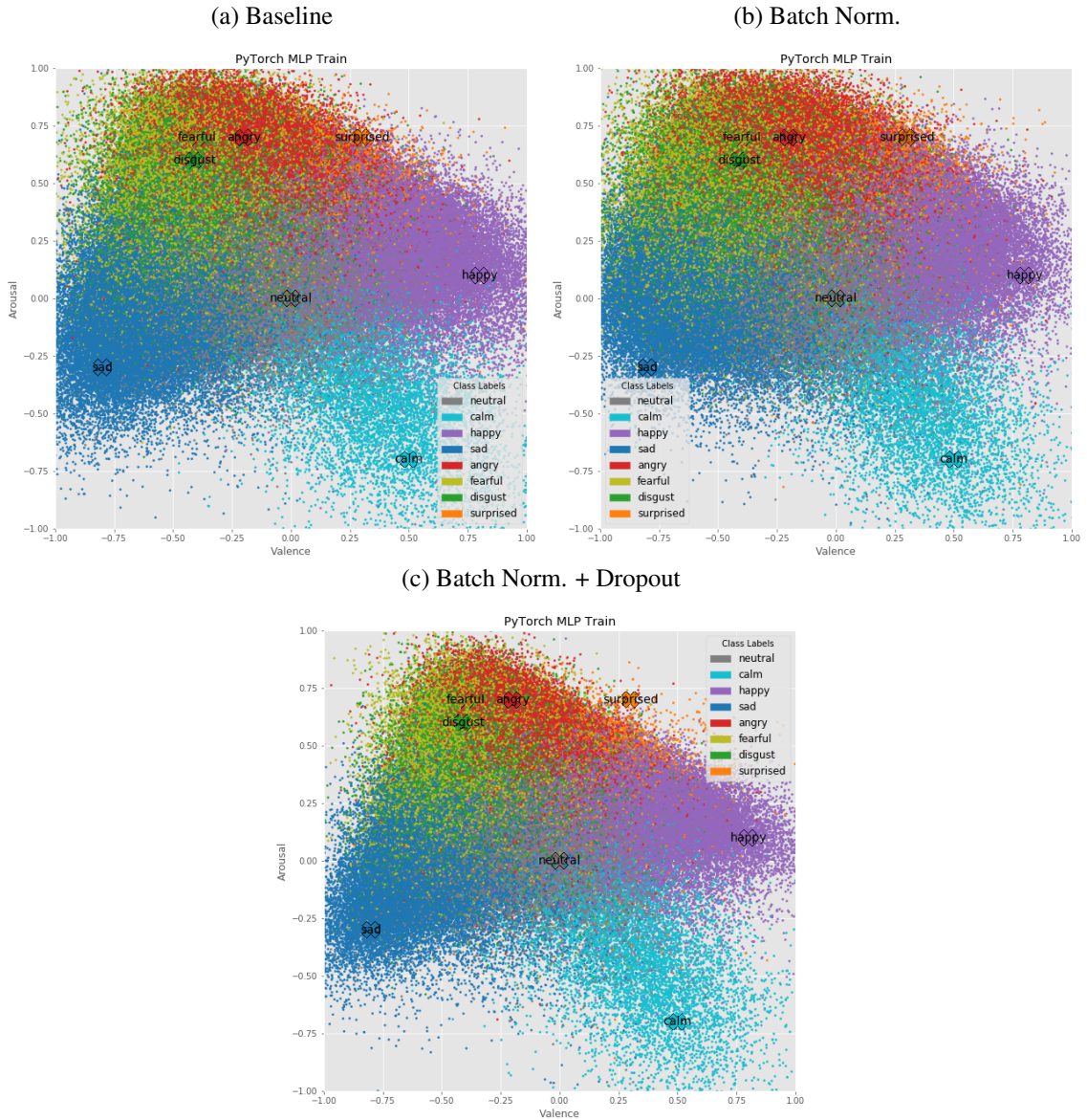
PyTorch MLP Configuration	Train	Validation	Test	Wild
Baseline	<b>0.123</b>	0.150	0.149	0.247
Batch Norm.	0.131	0.153	0.153	0.236
Batch Norm. + Dropout	0.127	<b>0.147</b>	<b>0.146</b>	<b>0.218</b>

We provide scores for the following configurations of our neural network: baseline performance, baseline with batch normalisation and finally the baseline with batch normalisation and dropout.

#### 4.2.2 Discussion

We can further analyse these results by looking at the distribution of predicted data points provided for the features of each FFT window. First we can look at the way the training data has distributed for each of our models.

Figure 4.1: Plotting training data on the circumplex model

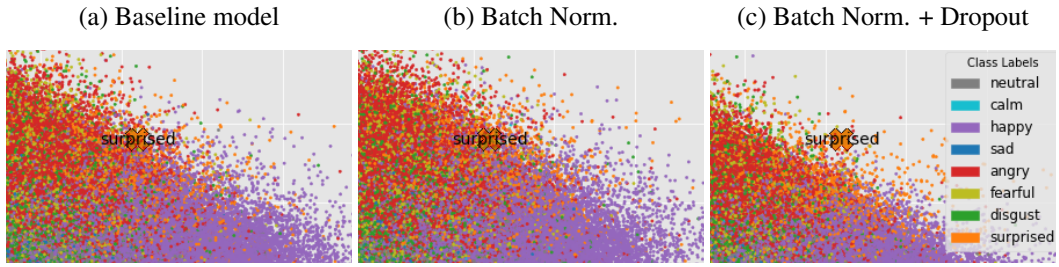


The clearest difference we can see between these distributions is the precision regarding clustering for each discrete class. Although the error is lowest for our baseline training (0.123), the spread of data points for both the baseline and baseline with batch normalisation models has a much broader spread around the ground truth labels. This is most clear when looking at the surprised samples. Notice the general mass of prediction data with our baseline w/batch normalisation and dropout model does not overlap the truth label, on close inspection the separation of classes is much better, which should result in less samples being wrongly predicted as the surprised class. It could be said that the model also has less confidence in some of the surprised samples also by this ruling.



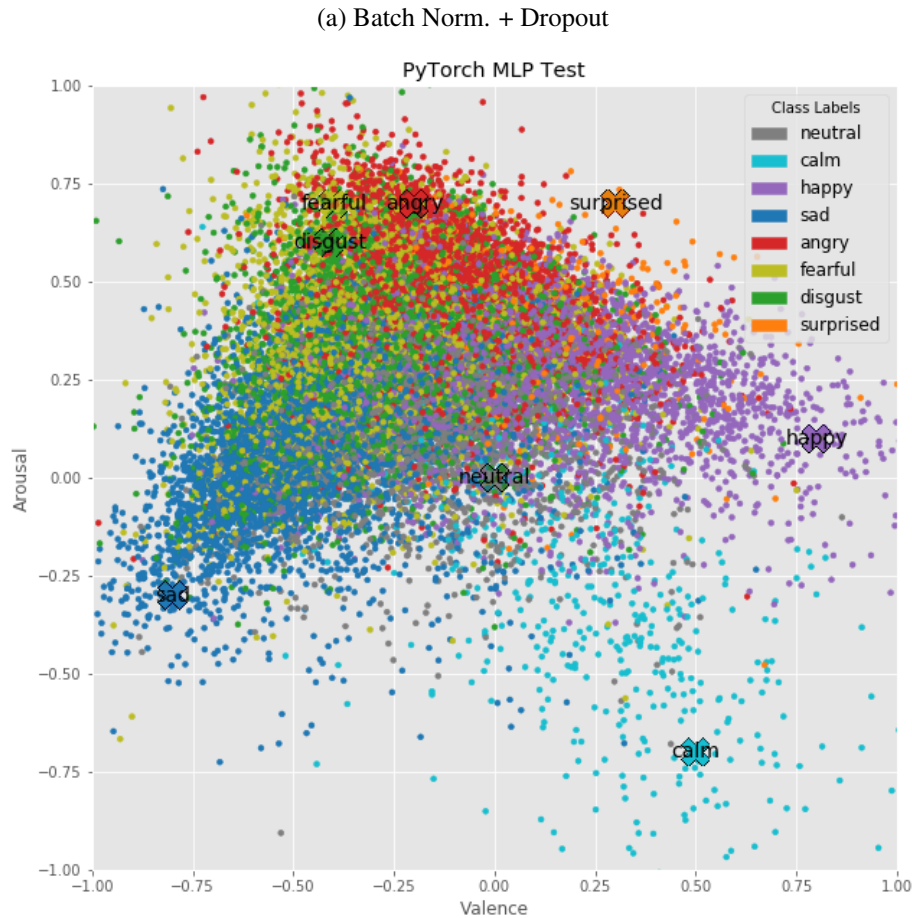
This trend continues throughout the validation and test prediction sets. Unfortunately it has not been possible to clearly observe good separation of the available classes to the top left of the circumplex model, angry, fear and disgust, in any of the model configurations. Due to the proximity of their ground truth coordinates, it is difficult to draw conclusions regarding optimal separation. However, based on minor visual differences and the error results provided in table 4.1, we can make the assumption that our batch normalisation and dropout combined approach is the optimal available model for predictions.

Figure 4.2: A closer look at distribution of training data on the circumplex model



The best result from our experimentation has come with the more than satisfactory performance of out test data from our initial train, validation, test split.

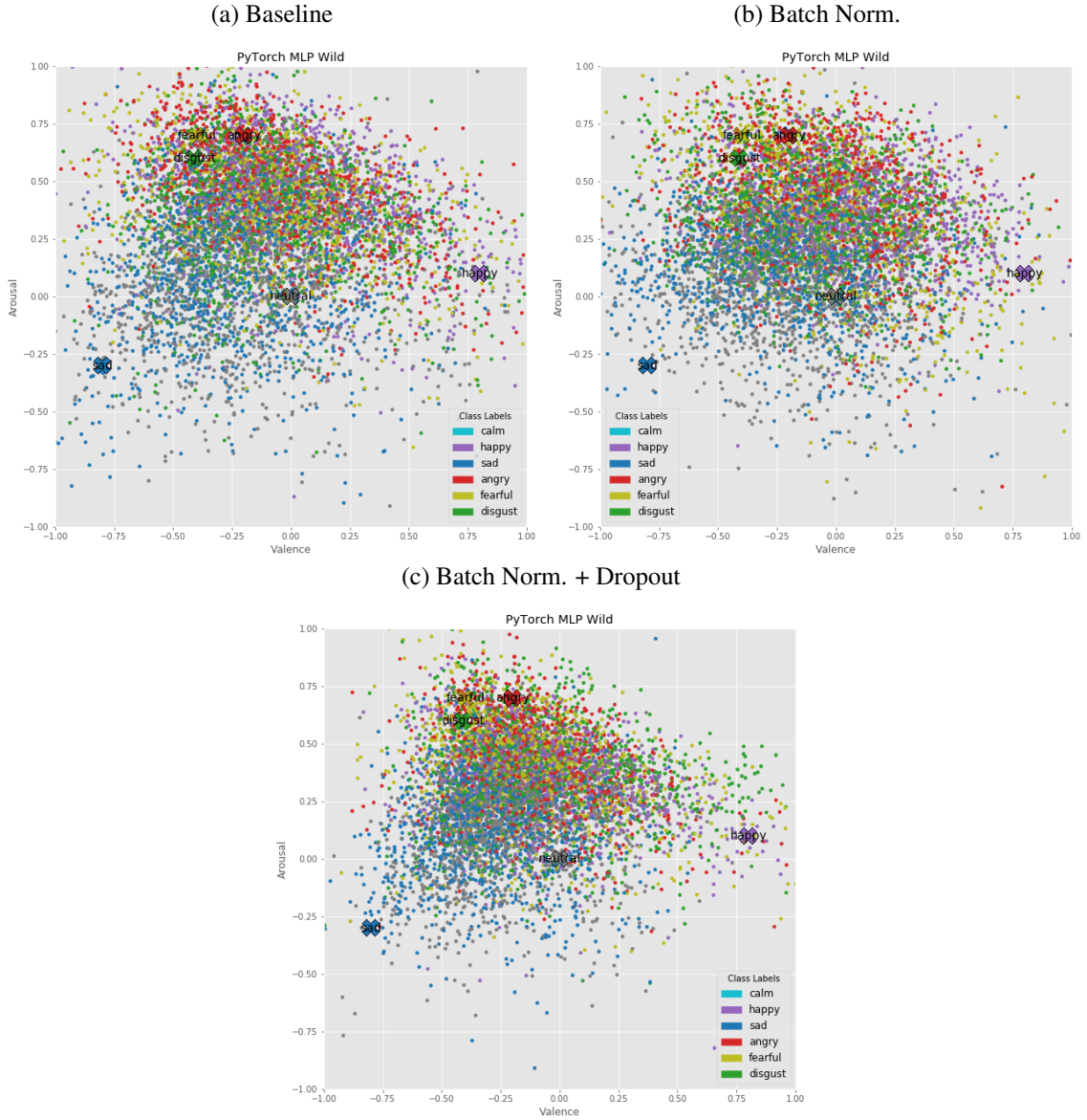
Figure 4.3: Plotting our test data on the circumplex model





At the outset of this research project, the main aim was to achieve a useful level of performance on real-world data. Currently this is the only aim of the paper which we have not fully achieved, however we can analyse those predictions on the circumplex model to determine a level of effectiveness, as well as the potential to threshold a level of confidence in our predictions to create a situation where predictions on unseen, non-structured data are useful.

Figure 4.4: Plotting our emulated wild data on the circumplex model



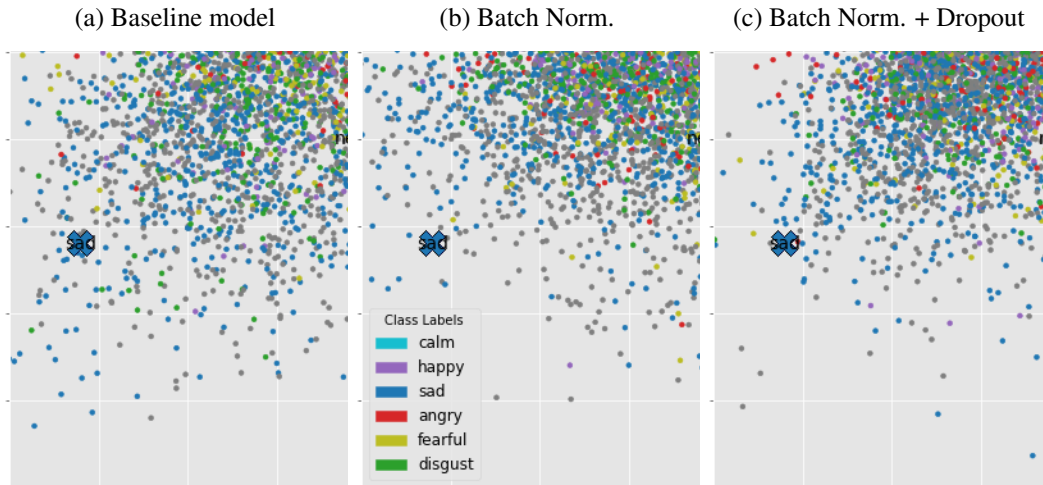
We can argue that the performance on data from some of the discrete emotion classes, particularly sad in this case, is very much at an acceptable standard. However there are clear issue regarding the available classes to the top left of the circumplex model, angry, fear and disgust, as well as lackluster performance with happy predictions vs. truth.

Looking at the overall distribution of our emulated 'wild' set a similar pattern to that of the training, evaluation and test predictions is shown, with more precise clustering around ground truth labels with our batch normalisation and dropout model, as well as less extreme values of valence and

arousal as a whole. There are significantly less outliers beyond the scope of our true X, Y coordinates, showing our model is becoming more effective in a use case where we employ a threshold nearest neighbour result to our predictions in deployment scenarios.

On closer comparison we can see a reduction of data points from prediction not belonging to sad being predicted incorrectly.

Figure 4.5: A closer look at distribution of emulated wild data on the circumplex model



The shape to predictions provided by the best performing model, that of batch normalisation and dropout being more shaped toward truth shows that it would likely be much more effective once all of the points in the circumplex model are taken into consideration as seen in figure 2.3.

## Chapter 5

## Conclusions

Initially, it was clear that speech-based emotion recognition could be used to provide measure, as well as have impact upon, the long term behaviours of users and produce affect on global user experience. This paper has set a groundwork for further research and development projects to build on these core concepts.

## 5.1 Summary of Results

Although the training, validation and test scores proved more than adequate for the purpose of use for SER with voice communications, our emulated wild test performance was not sufficient to make a strong case for this simple architecture to achieve a high degree of accurate predictions in a deployed situation currently.

A broader study on performance on 'real-world' data may prove the conclusions found with our emulated 'wild' test set TESS and experiments using other data-sets were inaccurate regarding potential deployed performance. However we can assume that a less performed vocalised emotion would be less likely to be predicted accurately on listening to the outliers with a high degree of prediction error.

## 5.2 Further Work

### 5.2.1 Deployment

Much is still to be done regarding deployment to compliment current in-game user behaviour metrics in Esports or as a moderation option for other forms of online voice communication. The practical investigatory work carried out over the course of this paper should now be deployed and tested over an active user base, with the resulting data used to further train and validate the model, which would gradually improve real-world test scores. With access to vast amounts of game-play metadata, development of a solution in industry would likely be extremely effective if well maintained.

### 5.2.2 Current deep learning methodology

It can be said that this paper lacks the inclusion of more complex neural network architectures, as well as more recent changes to the way audio feature selection and extraction is handled.

Our chosen feature set made up of MFCC is a popular approach for ASR work, which in the past was handled using Gaussian Mixture Models (GMMs) and Hidden Markov Models (HMMs). Since Deep Learning has moved to the forefront of data science activity, it can be questioned if MFCC are still the optimal feature set available. It is computationally possible now to remove the DCT step altogether and input an entire spectrogram to train a neural network. It is also beneficial to note that DCT is a linear transformation, and therefore can be sub-optimal in that it discards information in speech signals which are non-linear.

Given that the Fourier Transform itself is also a linear operation, it may be beneficial to ignore traditional pre-processing methodology and attempt to learn directly from the signal in the time domain, using a state of the art approaches such as a recurrent neural network (RNN) or long short-term memory network architecture (LSTM). This more complex approach could provide a more

optimal model for this SER task.

## Bibliography

- [AO73] Gordon Frederick Arnold and JD O'Connor. *Intonation of colloquial English*. Longman, London, 1973.
- [BK14] Jeremy Blackburn and Haewoon Kwak. Stfu noob! predicting crowdsourced decisions on toxic behavior in online games. In *Proceedings of the 23rd international conference on World wide web*, pages 877–888, 2014.
- [CCK<sup>+</sup>14] Houwei Cao, David G Cooper, Michael K Keutmann, Ruben C Gur, Ani Nenkova, and Ragini Verma. Crema-d: Crowd-sourced emotional multimodal actors dataset. *IEEE transactions on affective computing*, 5(4):377–390, 2014.
- [CDCT<sup>+</sup>01] Roddy Cowie, Ellen Douglas-Cowie, Nicolas Tsapatsoulis, George Votsis, Stefanos Kollias, Winfried Fellenz, and John G Taylor. Emotion recognition in human-computer interaction. *IEEE Signal processing magazine*, 18(1):32–80, 2001.
- [DAKD12] Mohit Dua, RK Aggarwal, Virender Kadyan, and Shelza Dua. Punjabi automatic speech recognition using htk. *International Journal of Computer Science Issues (IJCSI)*, 9(4):359, 2012.
- [Dav13] Namrata Dave. Feature extraction methods lpc, plp and mfcc in speech recognition. *International journal for advance research in engineering and technology*, 1(6):1–4, 2013.
- [DJV<sup>+</sup>19] Eero-Pekka Damskägg, Lauri Juvela, Vesa Välimäki, et al. Real-time modeling of audio distortion circuits with deep learning. In *Proc. Int. Sound and Music Computing Conf.(SMC-19), Malaga, Spain*, pages 332–339, 2019.
- [EAKK11] Moataz El Ayadi, Mohamed S Kamel, and Fakhri Karray. Survey on speech emotion recognition: Features, classification schemes, and databases. *Pattern Recognition*, 44(3):572–587, 2011.
- [HCCP06] Wei Han, Cheong-Fat Chan, Chiu-Sing Choy, and Kong-Pang Pun. An efficient mfcc extraction method in speech recognition. In *2006 IEEE international symposium on circuits and systems*, pages 4–pp. IEEE, 2006.
- [HE18] Zhaocheng Huang and Julien Epps. Prediction of emotion change from speech. *Frontiers in ICT*, 5:11, 2018.
- [IS15] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.
- [Jim] Jimo. Reporting to the overwatch. [Online; accessed Sep 19, 2020].

- [KB14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [KBH15] Haewoon Kwak, Jeremy Blackburn, and Seungyeop Han. Exploring cyberbullying and other toxic behavior in team competition online games. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 3739–3748, 2015.
- [KRB08] Aditya Bihar Kandali, Aurobinda Routray, and Tapan Kumar Basu. Emotion recognition from assamese speeches using mfcc features and gmm classifier. In *TENCON 2008-2008 IEEE region 10 conference*, pages 1–5. IEEE, 2008.
- [Lan18] Agnieszka Landowska. Towards new mappings between emotion representation models. *Applied Sciences*, 8(2):274, 2018.
- [LCHY19] Xiang Li, Shuo Chen, Xiaolin Hu, and Jian Yang. Understanding the disharmony between dropout and batch normalization by variance shift. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2682–2690, 2019.
- [Lea19] Anti-Defamation League. Free to play: Hate, harassment and positive social experiences in online games, 2019.
- [LR18] 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. *PloS one*, 13(5):e0196391, 2018.
- [Mik] Mikhail Ryazanov. Clipping waveform. [Online; accessed Sep 19, 2020].
- [MVM18] Belén Mesurado, Elisabeth Malonda Vidal, and Anna Llorca Mestre. Negative emotions and behaviour: The role of regulatory emotional self-efficacy. *Journal of adolescence*, 64:62–71, 2018.
- [NMS<sup>+</sup>18] Arun Narayanan, Ananya Misra, Khe Chai Sim, Golan Pundak, Anshuman Tripathi, Mohamed Elfeky, Parisa Haghani, Trevor Strohman, and Michiel Bacchiani. Toward domain-invariant speech recognition via large scale training. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 441–447. IEEE, 2018.
- [PFD20] M. Kathleen Pichora-Fuller and Kate Dupuis. Toronto emotional speech set (TESS), 2020.
- [RHW86] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.
- [Rus80] James A Russell. A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161, 1980.
- [Sch58] Maria Schubiger. *English intonation, its form and function*. M. Niemeyer Verlag, 1958.
- [Sch05] Klaus R Scherer. What are emotions? and how can they be measured? *Social science information*, 44(4):695–729, 2005.
- [SHK<sup>+</sup>14] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

- [Sof20] Valve Software. Squelching the noise, Feb 2020.
- [TC66] Tom W Tillman and Raymond Carhart. An expanded test for speech discrimination utilizing cnc monosyllabic words: Northwestern university auditory test no. 6. Technical report, Northwestern Univ Evanston Il Auditory Research Lab, 1966.
- [Tro11] Bård Tronvoll. Negative emotions and their effect on customer complaint behaviour. *Journal of Service Management*, 2011.