UTS UNIVERSITY OF TECHNOLOGY SYDNEY

Neural Networks for Music Emotion Recognition and Social Tags Emotion Representation

by Na He

Thesis submitted in fulfilment of the requirements for the degree of

Doctor of Philosophy

under the supervision of Sam Ferguson

University of Technology Sydney
Faculty of Engineering and Information Technology

April 2023

CERTIFICATE OF ORIGINAL AUTHORSHIP

I, *Na He* declare that this thesis is submitted in fulfilment of the requirements for the award of *Doctor of Philosophy*, in the *School of Computer Science*, *Faculty of Engineering & IT* at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution. This research is supported by the Australian Government Research Training Program.

Signature: Production Note: Signature removed prior to publication.

Date: April 2023

Acknowledgements

This work presents an important journey in my life, and it could not have been achieved without great support from many people. First, I would like to express my deepest gratitude to my supervisor Dr Sam Ferguson for his great help and guidance throughout my PhD. During these years, he inspired me a lot to do in-depth study in my research area and shared a large amount of his research experience to improve my research skills. When I was stuck in the research, he always encouraged me and recognized my work to build up my confidence and motivation. I am very fortunate to work with such a considerate and professional supervisor. I would also like to thank my co-supervisor Linchao Zhu and the members of the candidate assessment panel. They shared valuable comments and advice with me.

Then I highly appreciate the IT and lab staff who maintain my PC and iHPC lab to make my data source and development environment conducted with high efficiency. I also thank UTS Graduate Research School (GRS) and our faculty. They provide practical and high-quality seminars and workshops to enhance my research skills and the capabilities of preparing my thesis and keep my health mentally and psychologically. Moreover, our school and library provide comfortable research areas and lab sites with modern facilities that benefit our study, communication and experiment. Special thanks go to school academic officers Margot Kopel and Janet Stack. They dedicate themselves to administration work to inform us of school service, job opportunities, review reports, kinds of financial support and so on.

Finally, I must thank the APA scholarship for relieving my financial burden and our university for providing free literature access and a series of useful 3rd party tools, which mean a lot to me.

Na He

Sydney, Australia, 2022

List of Publications

Journal Papers

J-1. **He, N.** and Ferguson, S., 2022. Music emotion recognition based on segment-level two-stage learning. International Journal of Multimedia Information Retrieval, pp.1-12.

Conference Papers

C-1. **He, N.** and Ferguson, S., 2020, December. Music Social Tags Representation in Dimensional Emotion Models. In 2020 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom) (pp. 819-826). IEEE.

C-2. **He, N.** and Ferguson, S., 2020, December. Multi-view Neural Networks for Raw Audio-based Music Emotion Recognition. In 2020 IEEE International Symposium on Multimedia (ISM) (pp. 168-172). IEEE.

ABSTRACT

Music Emotion Recognition (MER), an important branch of Music Information Retrieval (MIR) systems, has become a very active research area, driven by the need to detect emotion in music automatically. A great deal of research has contributed to this area. With the emergence of neural networks, MER research has evolved from traditional machine learning methods combined with acoustic features to neural network learning methods combined with multi-source features. However, research gaps still exist in the following aspects. First, most existing research uses pre-processed audio features as learning model inputs, which require domain knowledge and work effort for feature selection. Limited attempts are made to use raw audio as model input directly. Secondly, few researchers partitioned the given music clips into shorter segments as model inputs due to the lack of segment-level target labels for supervised learning. Additionally, utilizing social tags is a good way to provide annotations for music emotion recognition. But tags are usually selected within a limited set of emotion corpus as discrete labels. Research rarely focuses on large-scale tags analysis and quantifies them in a dimensional emotion space.

I proposed solutions based on neural network methodologies to fill the above research gaps. Regarding the first point, I adopt a novel end-to-end deep learning architecture where multi-view convolutional neural networks are designed as feature extractors, followed by Bidirectional Long Short-Term Memory (BiLSTM) to capture temporal context sufficiently and predict dynamic music emotion. For the second one, I designed the two-stage learning framework, which uses music segments as model inputs without requiring segment-level labels. By applying the unsupervised learning method, segment-level feature representation could be generated. Then these time-series segment-level features are assembled and fed into a BiLSTM model to achieve the final music emotion classification. For the last one, I

contributed to social tag analysis related to music emotion by utilizing neural word embedding approaches. This way, social tags could be mapped into the dimensional emotion plane for further quantitative use.

To conclude, my research aims to improve the performance of music emotion recognition with neural network methods and study social tags representation for emotion annotation using word embedding techniques. This thesis presents all of the solution details. Meanwhile, music emotion background, related research work and plans are added to give a better view.

Contents

List of Figures

List of Tables

Abbreviations

ACT - Affective Circumplex Transformation

ANN - Artificial Neural Network

ANOVA - Analysis of Variance

BE - Backward Elimination

BiLSTM - Bidirectional Long Short-Term Memory

CBOW - Continuous Bag-Of-Words

CNN - Convolutional Neural Network

CFS - Correlation-based Feature Selection

DBM - Deep Boltzmann Machine

DBN - Deep Belief Network

DTM - Document-term matrix

EDA - Electrodermal Activity

EEG - Electroencephalogram

FFT - Fast Fourier Transform

FS - Forward Selection

GAN - Generative Adversarial Network

GAT - Graph Attention Network

GMMs - Gaussian Mixture Models

GNN - Graph Neural Network

GRU - Gated Recurrent Unit

KNN - K-Nearest Neighbors

LASSO - Least Absolute Shrinkage and Selection Operator

LDA - Linear Discriminant Analysis

LSA - Latent Semantic Analysis

LSTM - Long Short-Term Memory

MCRNN - Multi-view Convolutional Recurrent Neural Networks

MER - Music Emotion Recognition

MFCC - Mel-frequency Cepstral Coefficient

MIR - Music Information Retrieval

MIREX - Music Information Retrieval Evaluation eXchange

MLR - Multiple Linear Regression

MP3 - MPEG layer 3

MSD - Million Song Dataset

MSE - Mean Square Error

MSS - Music Source Separation

NLP - Natural Language Processing

nMDS - non-metric multidimensional scaling

PA - Procrustes Analysis

RBM - Restricted Boltzmann Machine

ReLU - Rectified Linear Unit

RMSE - Root Mean Square Error

RNN - Recurrent Neural Network

SCF - Spectral Crest Factors

SFM - Spectral Flatness Measures

SONE - Specific Loudness Sensation Coefficients

STFT - Short-time Fourier Transform

SVD - Singular Value Decomposition

SVM - Support Vector Machine

TF-IDF - Term Frequency-Inverse Document Frequency

VA - Valence-Arousal

VAD - Valence-Arousal-Dominance

VSM - Vector Space Model

ZCR - Zero-Crossing Rate

Chapter 1

Introduction

In the digital information age, facing an enormous amount of online music resources, Music Information Retrieval (MIR) plays a significant role in retrieving information. Music Emotion Recognition (MER) is one fast-growing branch of MIR. It benefits emotion-based music applications and improves personality experiences through music psychology, recommendation systems and artificial intelligence. Generally, MER research consists of 4 parts: music data collection, emotion model definition, music feature extraction and MER model design. First, the music dataset is collected. It typically includes music samples (usually audio signals) and their emotional annotation (tags or ratings). Meanwhile, a dimensional or categorical emotion model is defined to annotate music. Then, music features are extracted from one or more sources like raw audio and lyrics. After that, feature data is fed into the designed learning model to train emotion prediction patterns, usually with emotion labels as targets. Once the model finds the optimal pattern based on the evaluation matrix, it could be used to recognize music emotion for other unlabelled music.

Before the widespread use of neural network approaches, traditional machine learning models were mainly utilized to solve emotion classification or regression problems, those models usually have less ability to represent features. So the performance relied on feature collection before model training to a great extent. Considering this limitation of those models, most researchers tended to design and extract human-engineered audio features as model inputs to gain better performance (Schmidt et al., 2010; Panda et al., 2018). In recent years, the rapid development of deep learning has brought MER into a new stage. With layers of neural networks, these models can learn music features automatically from raw data or low-level music features. Due to this, researchers prefer to dedicate themselves to designing

efficient model architectures (Choi et al., 2017; Dong et al., 2019), or explore the combination of multiple feature sources (such as audio, lyrics, social tags and electroencephalogram signals) with adopting multimodal fusion strategies (Lian et al., 2018; Hu et al., 2017).

Regarding emotion annotation, it could be conducted by subjective test, but this usually results in a heavy load on time consumption and labour cost (Yang and Chen, 2012), which is not tractable with large-scale datasets like those seen in MIR. Instead, increasing interest has been shown in crowdsourcing resources (Çano and Morisio, 2017a). With the fast growth of web social media, social tags from community users are viewed as a good source to provide annotation for music-related tasks such as music auto-tagging (Choi et al., 2016), music emotion recognition (Delbouys et al., 2018) or sentiment analysis (Çano and Morisio, 2017b). Social tags save more effort than subjective annotation and serve training models better for large-scale datasets.

This thesis focuses on music emotion research. The research aims to improve the performance of music emotion recognition with deep learning methods and study emotion representation from music social tags by using word embedding techniques. Such emotion recognition will contribute to emotion-based music categorization and retrieval. Further, some personalized music applications could combine organized music and user-oriented data to select and recommend music based on emotion to improve the listening experience and even music therapy. The rest of this chapter presents the motivation, research questions, research objectives, results and contributions, as well as the outline of this thesis.

1.1 Motivation and Research Questions

MER research has reached great achievements in many aspects. But issues and limitations still exist mainly due to the subjectivity and vagueness of emotional response. Moreover, researchers got used to reusing the common frameworks formed in previous MER research. That leads to a lack of insights into some study points. First, most researchers use pre-processed audio data rather than raw audio for train-

ing models. Especially before the emergence of deep learning, performance was mostly improved through better hand-engineered audio features. (Laurier et al., 2010; Schmidt et al., 2010). But such feature extraction work usually requires professional acoustic domain knowledge and operating cost, even though many musical phenomena are not yet defined or understood in a unified way. If we could find a way to use raw audio directly and make emotion predictions efficiently, then much time and labour costs would be saved. Second, much research naturally adopts supervised learning methods for labelled data. They usually used the given music clips as model inputs to train the learning pattern and match the prediction with the given target labels. There are few good methods to partition the given music clips into shorter segments as model inputs due to the lack of segment-level target labels, even if the duration of some music clips is relatively long and unsuitable for analyzing emotion cues. In addition, previous research usually maps social tags to the quadrants of the classic dimensional emotion model, then define annotation schemes to project songs associated with tags to dimensional space (Panda et al., 2018), rather than analyzing tags relationships in the context of music tag dataset. Only a few research projects focused on tags analysis to reflect themselves on dimensional emotion model (Saari and Eerola, 2014; Laurier et al., 2009)). Still, these approaches use conventional latent semantic analysis (LSA), which depends on geometrical transformations rather than neural text analysis. I pay attention to these research gaps and expound my corresponding solutions in this thesis.

Based on the facts mentioned above, my research questions are proposed:

1. How to use raw audio rather than human-engineered features to predict music emotion?

2. How to use segment-level music recordings to predict music emotion without requiring extra annotations?

3. How to analyze and reflect on the relationship between social tags and music emotion?

1.2 Research Objectives

According to the research questions, the research objectives are set up as follows:

i. Design a deep neural network model that can directly use raw audio signal data as training inputs rather than using pre-processed audio features and achieve better performance for emotion prediction.

ii. Propose a deep learning architecture that could accept segments partitioned from the given music clips as model inputs without collecting extra segment-level annotations and complete final emotion recognition.

iii. Provide a solution for music social tags analysis with neural word embedding models applied to represent tags in a dimensional emotion space.

1.3 Results and Contributions

The main contributions of my research work are presented below,

- A multi-view CNN (Convolutional Neural Network)-based model is designed. It can accept raw audio as inputs and extract music features from multiple perspectives automatically and efficiently. Then these features are aggregated and fed into Recurrent Neural Networks (RNNs) to learn time-varying information for dynamic emotion variation. Based on this structure, music emotion could be detected without expending too much effort on prior knowledge learning and feature extraction. At the same time, the performance gained 4% and 16% average improvement at RMSE and R^2 matrices compared with the previous model design.

- A segment-level two-stage MER architecture is proposed. It combines unsupervised learning as a feature extractor and supervised learning as an emotion recognizer. In the unsupervised learning module, a CNN-based autoencoder is employed to represent segment-level audio data without considering labels. Then, in the supervised learning stage, feature representations for time-series

segments are fed into appropriate RNNs to complete the final emotion prediction. In this way, the music pieces could be further partitioned into appropriately short segments without requiring extra annotations for model training. From the perspective of data augmentation, segment-level music increases the data scale and data variation for unsupervised learning, thereby enhancing the model performance. Compared to the previous model that used only music sources, my model performance achieved more than 10% increases in valence and arousal based on accuracy and F1-score. My model even competed with the latest multimodal framework.

- The text analysis solutions are applied to large-scale music social tags to represent tags in dimensional emotion space. In tags analysis, tags are viewed as terms rather than single words because tags may be phrases or sentences. The social tags dataset is preprocessed to generate structured inputs such as a text corpus or a factorized matrix for the subsequent tag analysis models. Then the neural text embedding models are trained and output vector-based terms. After a series of dimensional transformations, social tags could be represented quantitatively and show their relationship with each other in a dimensional emotion space.

Those contributions have been published in journal and conference papers. Refer to the "List of Publications" page for details.

1.4 Thesis Organization

In this thesis, my work focuses on two scenarios for music emotion recognition and one related study of social tags representation for music emotion. The first scenario is detecting dynamic emotion variation by using raw audio signals. It aims to solve a regression problem referring to the first research objective. The other scenario focuses on static emotion classification to achieve the second research objective. The study of social tags for emotion representation commits to the third objective. This thesis is organized as follows:

- *Chapter 2*: This chapter presents the background knowledge about music emotion research, including emotion definition, emotion taxonomy, music features and dataset introduction.

- *Chapter 3*: This chapter is the literature review about music emotion recognition and social tags analysis. In detail, it includes feature selection and impact, emotion response time and segmentation, traditional machine learning methods, deep learning methods, multimodal strategies, social tags applications and cross-domain methodologies.

- *Chapter 4*: This chapter elaborates on the deep learning model for dynamic emotion detection using raw audio to achieve the first objective. The introduction section identifies the problems in the previous research and the superiority of my research. The methodologies part describes the multi-view model architecture and introduces the augmentation methods for raw audio data. Then the experimental details are given, followed by result analysis and some discussion.

- *Chapter 5*: This chapter elaborates on the deep learning model for static emotion classification to achieve the last two objectives. First, research gaps are pointed out, and my research contribution is stated. Then the segment-level two-stage model structure is presented with time and frequency masking method included. Following this, experimental details are given, including introducing two datasets, processing audio data, transforming annotation and setting up model training. The results are shown based on different segment levels and different baseline models. The performance analysis in multiple aspects is provided in the discussion part.

- *Chapter 6*: This chapter presents social tag analysis work for music emotion. The current research demand and my study are introduced briefly. Then, based on a large-scale music tags dataset, methods for tag representation are described, including data preprocessing, tag embedding, emotion vector extraction and data transformation. Several tag embedding models are con-

ducted in the experiments, and their performance is compared and visualized.

- *Chapter 7*: A summary of the thesis contents is given in this chapter. Future works are listed as well.

Chapter 2

Music Emotion and Music Features

Music plays a significant role in human life. With the rapid development of digital technology, music recordings have transformed from physical materials to online resources. Facing a huge amount of digital music information, the research for music information retrieval has made great progress. Initially, music was usually retrieved based on catalogue metadata such as song title, artist name, album name and genre. Then, to improve the personalized experience, diversified user-oriented systems and applications have been proposed. In this procedure, the fast-growing demand of music organization by emotion has gained more attention (Yang and Chen, 2012).

This chapter introduces the background of music emotion from the following aspects.

- the concept, scope and taxonomy of music emotion
- the introduction of music features and related tools for feature extraction
- music feature selection and impact on emotion
- the overview of music-related datasets

2.1 Music Emotion

This part discusses the concept of music emotion and indicates the emotion type my thesis focuses on in MER research. On the other hand, a series of typical emotion definition models are illustrated.

2.1.1 Music Emotion Introduction

What is emotion? From a psychological perspective, emotion is an immediate physiological response to a perceived stimulus. Sometimes, we view other terms as synonyms with emotion and use them interchangeably, such as feeling, mood, affect, and sentiment. But Ketai (1975) pointed out that such words refer to distinct psychological phenomena and should be used carefully. Feelings cover a larger scope of subjective experience beyond emotions (Scherer, 2005). It includes the physical and mental sensations, not only emotional experiences. A mood is an affective state in psychology. In contrast to emotions, moods are less specific, less intense and less likely to be provoked or instantiated by a particular stimulus or event (Beedie et al., 2005). Affect is an umbrella term that embodies both emotion and mood. Emotions are one class of expressions of affect. A sentiment involves both a physiological reaction and a cognitive, subjective component. Unlike emotions, sentiments are enduring dispositions targeted toward an object (Munezero et al., 2014). Moreover, these researchers also pointed out that the key factors of distinguishing them are response time and duration. Compared with moods, feelings and sentiments, emotions come first and have a relatively shorter duration. Regarding the affect you are experiencing from music, my study focuses on music emotion research.

Further, we need to confirm the type of emotion in the study of music emotion recognition. Generally, music emotions could be divided into expected, perceived, and induced emotions (Zhao et al., 2019). Expected emotion is the emotion that the music creator intends to convey, while the latter two refer to the emotional response from listeners. The perceived emotion means what kind of emotion people perceive in music, whereas the induced emotion (also known as felt emotion) is the emotion actually experienced by the listeners. Kallinen and Ravaja (2006) argued that induced emotion is more subjective than perceived emotion. Thus, the level of agreement among listeners for perceived emotion is higher. On the other hand, Song et al. (2013) showed that the gap in the agreement for both emotions is small. To cover common situations, MER researchers tend to focus on perceived emotion.

In studying emotion perception, we must confirm that the research focuses on

detecting stable or dynamic emotions. That relies on the research purpose and the characteristics of music datasets. This thesis covers both scenarios and uses different deep-learning models to detect either a single emotion or emotion variation.

2.1.2 Emotion Taxonomy

Regarding the subjectivity of human emotion and the difference in culture and living background, it is difficult to define and encompass all human emotions. Even so, there are two typical taxonomic approaches that most researchers follow: the categorical approach and the dimensional approach. The categorical approach maps emotion descriptions into some typical discrete categories. The complexity of annotation and classification is relatively low. Still, the semantic gap exists as a limited set of terms restricts the coverage of the full range of human emotion, and even people show different understandings of one term. By contrast, the dimensional approach considers emotion as continuous values within a two- or three-dimensional space, where emotion description is not discrete terms but ratings or numbers in some metrics. So dimensional models are considered better approaches to avoid ambiguity issues, despite the relatively higher cost of gaining quality annotation and pattern recognition. Increasingly, much research tends to use a dimensional model (Grekow, 2018) or quadrants in dimensional space (Panda et al., 2018) to narrow the semantic gap between music experience and human perception. In MER tasks, we usually view the categorical model as a classification problem to predict the emotion classes. In contrast, the dimensional model is regarded as a regression problem to predict values.

In the following parts, some typical categorical and dimensional models are introduced. Those models reflect the main evolution of music emotion taxonomy.

Categorical Emotion Models

For the categorical models, they are usually made up of either several basic terms (Laurier et al., 2010) or categories/clusters of terms (Hu et al., 2009; Bhattacharya and Kadambari, 2018).

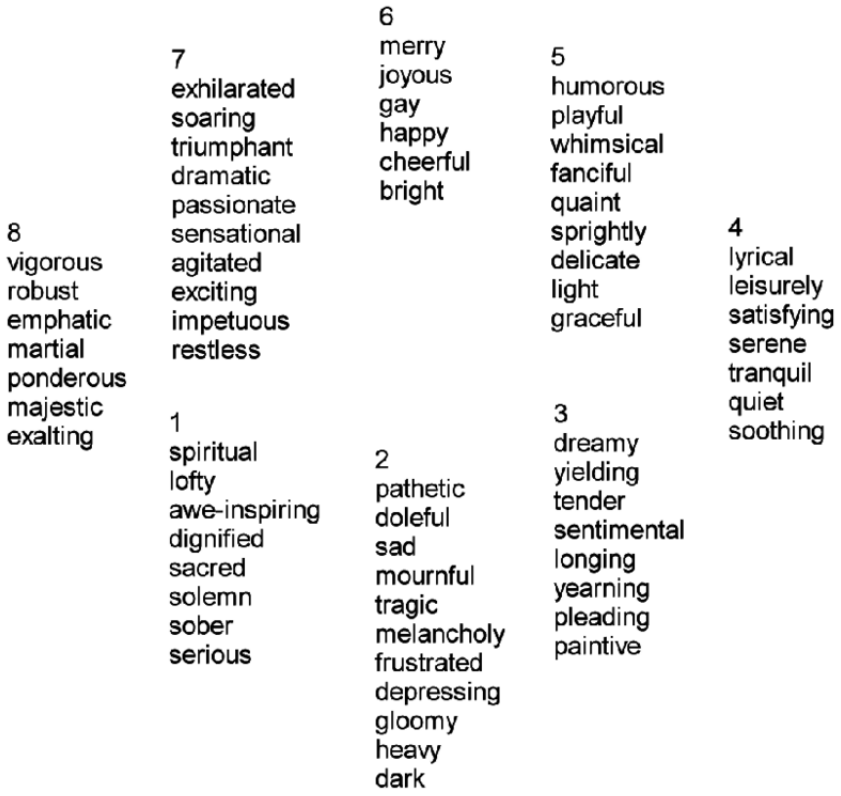


Figure 2.1 : Hevner's eight clusters of affective terms (Hevner, 1936)

The emotion model description in the early stage could be traced back to the paper written by Hevner (1936). She conducted experiments with approximately 450 subjects and concluded eight adjective clusters of affective terms laid out in a circle (see Figure 2.1). The adjectives within one cluster are closely related and compatible. While the adjacent clusters have some characteristics in common, the meaning between them varies in a cumulative way until the clusters in the opposite position have no likeness.

In recent decades, music emotion research has achieved great progress. One typical categorical model was proposed by Hu and Downie (2007). It consists of 5 mood clusters (presented in Table 2.1) collected from the `AMG` (now known as `AllMusic` *) mood repository. This model was adopted widely in mood classification tasks initiated by the Music Information Retrieval Evaluation eXchange (MIREX)

*https://www.allmusic.com/

campaign. On the other hand, benefited from the music social tags provided by Last.fm[†], some categorical models were derived from that large-scale dataset, such as 18 mood categories for lyrics text mining (Hu et al., 2009).

Table 2.1 : The MIREX Mood Classification

Categories	**Labels**
Cluster1	passionate, rousing, confident, boisterous, rowdy
Cluster2	rollicking, cheerful, fun, sweet, amiable/good natured
Cluster3	literate, poignant, wistful, bittersweet, autumnal, brooding
Cluster4	humorous, silly, campy, quirky, whimsical, witty, wry
Cluster5	aggressive, fiery, tense/anxious, intense, volatile, visceral

With the change of the times, the meaning of some words has been changed, such as 'gay' used in Hevner's model. Also, acoustic and semantic overlaps across clusters existed in the MIREX mood dataset (Panda et al., 2015). Such ambiguity makes it difficult to guarantee annotation consistency.

Dimensional Emotion Models

Regarding the dimensional models, the most well-known one was articulated by Russell (1980), who proposed 28 affect words located in a 2-dimensional circumplex model (see Figure 2.2), with finally evolved into a horizontal axis of valence (pleasure-displeasure) and a vertical axis of arousal (active-inactive) (Russell and Barrett, 1999). By contrast, Thayer (1989) proposed another dimensional model based on tension and energy. These two models could quantitatively describe emotion from different aspects (see Figure 2.3). Based on this, Scherer (2005) tried to integrate them into a semantic space for measuring emotion-related social labels (see Figure 2.4), which benefits much social information analysis regarding emotion (Paltoglou and Thelwall, 2013; Saari and Eerola, 2014).

[†]http://www.last.fm

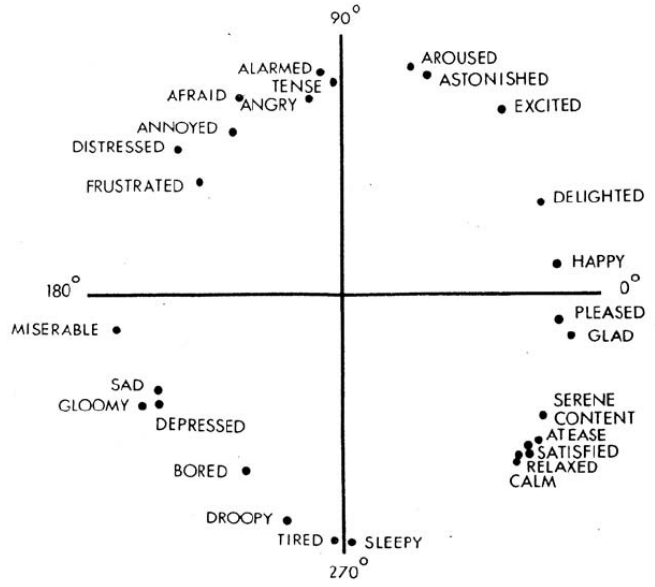


Figure 2.2 : Russell's circumplex model of affect (Russell, 1980)

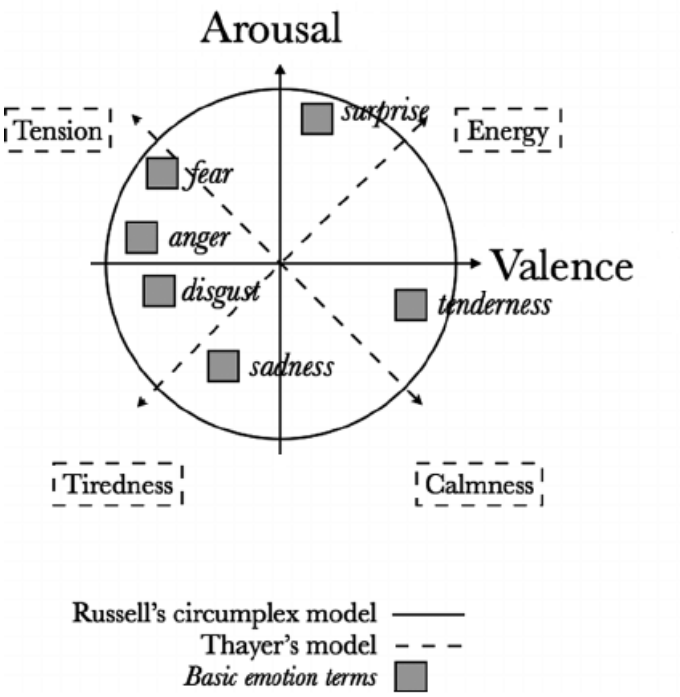


Figure 2.3 : Schematic diagram of the combination of Russell's and Thayer's models (Eerola and Vuoskoski, 2011)

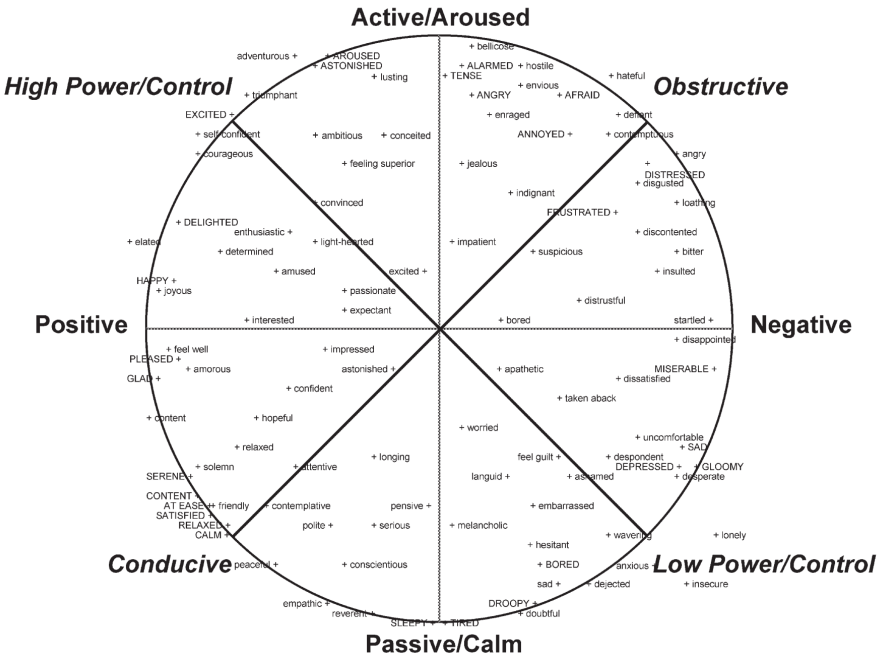


Figure 2.4 : Scherer's semantic space for emotion (Scherer, 2005)

On the other hand, 3-dimensional emotion models were designed as the extension of the valence-arousal (VA) model. Schimmack and Grob (2000) proposed energetic and tense arousal combined with valence and demonstrated that these three dimensions could not be reduced into two dimensions. Another famous 3-dimensional model is the valence-arousal-dominance (VAD) model (see Figure 2.5), where dominance means the degree of control ranging from controlled to in control. This model could distinguish some emotion types more clearly. For example, 'anger' and 'fear' are usually located in the same quadrant of VA space due to dimensionality limitation, but we could see their difference intuitively through the dominance dimension.

Further, researchers developed some categorical models based on these dimensional spaces, especially the 2D VA emotion model. They used either four quadrants with representative terms in each quadrant or high/low levels of VA. Some typical dimension-based categorical models are shown in Table 2.2

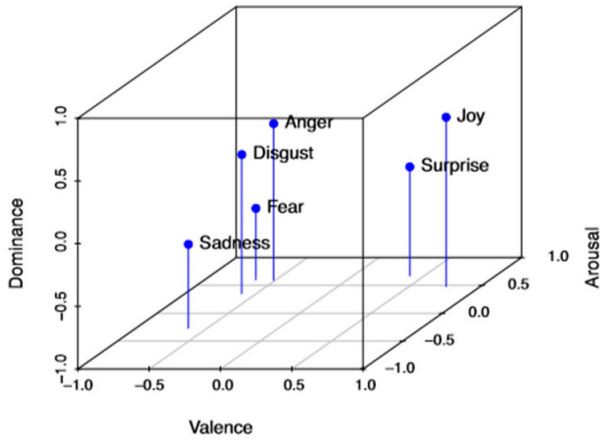


Figure 2.5 : The Valence-Arousal-Dominance (VAD) 3-dimensional model (Bălan et al., 2020)

Table 2.2 : The Dimension-based Classification

Related research	**Categories**
Eerola and Vuoskoski (2011)	happiness, sadness, tenderness, anger and fear
Laurier et al. (2009)	4 clusters located in four quadrants
Panda et al. (2018)	quadrants based on dimensional space : Q1, Q2, Q3, Q4
Chung and Yoon (2012)	high/low arousal; high/low valence

2.2 Music Features

In MER tasks, a machine learning model is usually utilized to find the pattern between music features and emotion labels, thereby using it to predict unlabelled data. To do that, the first thing researchers need to determine is what kind of features are used. According to the feature sources that current studies could obtain, music features can be mainly classified into categories.

- **Content-based audio features:** These features are usually extracted from raw audio signals or represented from low-level features through audio analysis tools, such as loudness, pitch, timbre and tempo.
- **Lyrics:** This is another important content-based feature to assist emotion detection through text analysis (Hu et al., 2009). However, this feature is not applicable to some music, like instrumental music.
- **Context-based information:** This usually includes music context such as artist, title, album, genre, similarity, social tags and even user profile. Besides that, biological signals responding to music is used increasingly in recent years.

2.2.1 Audio Feature Introduction

Before starting up an MER research, it is better to learn about some acoustic features of music. Basically, many features could be classified into the time domain and frequency domain (spectral, cepstral and phase), but not limited to these domains. Some research also organized features according to their physical or perceptual basis (Alías et al., 2016). Yang and Chen (2012) review some features utilized in MER, including energy, rhythm, melody and timbre. Each of them represents a group of specific related features. Song et al. (2012) selected 55 features and categorized them into four dimensions: dynamics, rhythm, spectral and harmony. Then they evaluated the feature impact on the performance of emotion classification. Grekow (2017) grouped music features as three sets (low-level features, rhythm and tonal) and checked the effect of feature combinations on the performance of detecting valence and arousal. Panda et al. (2018) summarized eight categories

of standard audio features. That is, melody, harmony, rhythm, dynamics, tone colour (or timbre), expressive techniques, musical texture and musical form. Then they proposed novel features that assist emotion recognition. Based on previous work, Vatolkin and Nagathil (2019) also outlined features commonly used as energy, harmony, rhythm and timbre. Generally, most features mentioned above could be extracted from signal analysis tools or packages (see Table 2.3) and be viewed as human-engineered features. Due to many music features varying with time, mean and standard deviation are often used to calculate the total metrics for music. As we can see, some categories are identified frequently. My thesis briefly introduces these common emotionally-relevant musical features in the following part.

Table 2.3 : Music Audio Feature Analysis Tools

Tool	**Related Literature or Reference**
Marsyas	(Tzanetakis and Cook, 2000)
MIRToolbox	(Lartillot et al., 2008)
PsySound	(Cabrera et al., 2007)
openSMILE	(Eyben et al., 2013)
Essentia	(Bogdanov et al., 2013)
librosa[‡]	(McFee et al., 2015)

Energy

Energy usually means audio power that is caused by object vibration. In the music domain, it usually represents a perceptual measure of intensity and activity and is related to arousal. It also refers to the characteristics of sound in terms of pitch, volume and frequency. In some MER research (Song et al., 2012; Panda et al., 2018), they named dynamics to cover similar features. One of the typical energy-related features is loudness. Loudness is the quality of a sound that is the primary

[‡]https://github.com/librosa

psychological correlate of physical strength (amplitude). In some audio analysis tools, they could be further extracted as total loudness, specific loudness sensation coefficients (SONE) and so on (Zhang et al., 2017). Vatolkin and Nagathil (2019) mentioned Zero-crossing rate (ZCR) as an energy feature representing the rate at which the audio signal crosses the zero amplitude level in a certain interval. These energy features are usually measured by 'high' or 'low'.

Rhythm

Rhythm is the pattern of music over time and is the one indispensable element for all music. Specifically, it reflects the regular notes/pulse/beats/meter changes over time. It is mainly described through tempo and beat. Tempo means the speed or pace of the music, measured by 'fast' or 'slow'. Beat means the underlying pulse in the music, measured by 'strong' or 'weak'. Besides that, other rhythm-related features are mentioned in some research work, such as fluctuation (Song et al., 2012), rhythm strength, rhythm regularity, rhythm clarity (Yang and Chen, 2012), onsets (Panda et al., 2018).

Timbre

Timbre, also known as tone colour or tone quality, is the perceived sound quality of a musical note, sound or tone. It distinguishes different types of sound sources, like musical instruments or human voices (even for the same note) or different instruments in the same category. It is mainly related to frequency domain features, such as cepstral timbre, spectral and phase domain timbre. A commonly used timbre feature in MER is Mel-frequency cepstral coefficients (MFCC). It is a type of cepstral representation and reflects the average of the spectral distribution. However, such averaging leads to the loss of spectral information. Due to this, Octave-based spectral contrast was adopted to roughly reflect the relative distribution of the harmonic components in the spectrum (Schmidt et al., 2010). Depending on the audio analysis tools researchers used, they provided some spectral features from different aspects. MIR toolbox could extract features related to the sensory dissonance of the music, such as roughness, irregularity and brightness. In comparison, Marsyas could

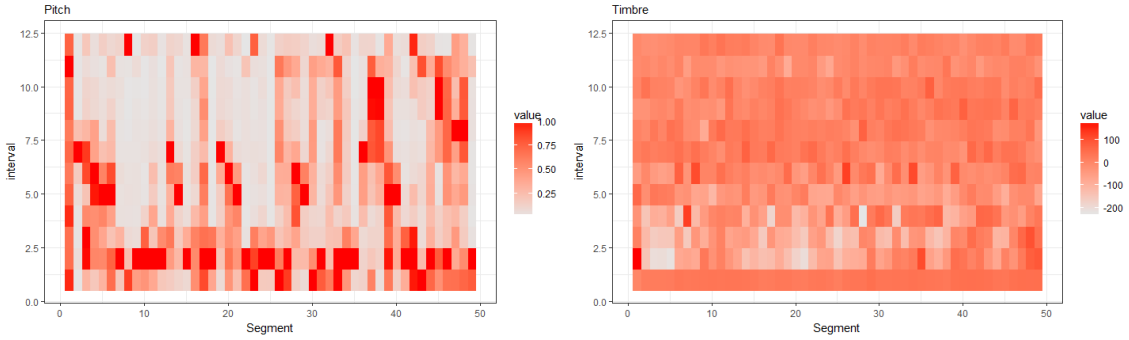


Figure 2.6 : Heat map for pitch and timbre[§]

extract features related to the noisiness of audio signals, such as spectral flatness measures (SFM) and spectral crest factors (SCF). Moreover, much research classified ZCR into this music dimension (Saari et al., 2011; Song et al., 2012; Panda et al., 2018), rather than energy feature.

Melody

Melody is a linear succession of musical tones that the listener perceives as a single entity. It implies rhythmically ordered movement from pitch to pitch so that melody features could be mainly identified as pitch. Pitch is a perceptual property of sounds that allows their ordering on a frequency-related scale, usually measured by 'high' or 'low'. Based on the twelve-note chromatic scale, Figure 2.6 shows an example of how the strength of pitch organizes in a music piece, contrasting with timbre to give an intuitive sense.

Harmony

Harmony is another perceptual property of music, along with the melody. The study of harmony involves chords, which means more than one note is played simultaneously. Harmony is often viewed as the "vertical" aspect of music, which is distinguished from melody and rhythm viewed as the "horizontal" aspect. It also

[§]Pitch and timbre data shown here is obtained from Million Song Dataset. For more details, please refer to https://developer.spotify.com/documentation/web-api/reference//operations/get-audio-analysis

refers to the concept of consonance and dissonance of chords, which impacts the emotional perception of music in opposites. Harmony-related features commonly used include chromagram, inharmonicity, key clarity, sharpness, etc.

Spectrogram

A spectrogram is a visual representation of the spectrum of frequencies of a signal as it varies with time. It is computed through the short-time Fourier transform (STFT). Considering the limited range of frequencies and amplitudes perceived by humans, the spectrogram is usually further transformed to mel scale to form the mel spectrogram. Its scaling is analogous to the range of human hearing. Inspired by image processing in many deep learning models, spectrogram or mel spectrogram is considered as 2-dimension audio feature input fed to convolutional neural networks (Bhattacharya and Kadambari, 2018; Nayal et al., 2019). In order to leverage the learning capability of such neural networks, I adopted spectrogram as music feature inputs in one scenario of my research so that I could focus on optimizing model design rather than feature selection.

2.2.2 Lyrics and Context-based Features

Lyrics is another important content-based feature source in MIR research (Hu et al., 2009). Lyrics analysis is covered by the mature research area of Natural Language Processing (NLP). It could reflect the song's main idea, but it is not music's inherent attribute, thereby lacking universality. For instance, it is not applicable to instrumental music. Based on text analysis, social media information such as user comments or social tags also contribute effort to some MIR applications, especially for auto-tagging (Choi et al., 2016) and emotion classification (Lin et al., 2011; Çano and Morisio, 2017b). As the crowdsourcing data, social tags or comments could provide large-scale music annotation contrasted with the subjective test, but the work effort for data cleaning is relatively high to solve synonymy, polysemy and noise, even malice and bias (Lamere, 2008; Saari and Eerola, 2014). Apart from that, song metadata (such as artist, title, album and genre) is a good source to build up music similarities and benefit music classification (Hu and Downie, 2007) and

recommendation (Han et al., 2010).

In recent years, many attempts have been made by using physiological signals like electroencephalogram (EEG) signals (Tripathi et al., 2017) or Electrodermal Activity (EDA) signals (Yin et al., 2020). Especially for affect research, traditional methods often took advantage of verbal expressions or non-verbal behaviour such as facial expressions and body gestures. The physiological responses provide new cues to improve emotion estimation.

2.3 Audio Feature Selection and Impact on Emotion

As content-based music features, audio features play a critical role in emotion recognition. Generally, music emotion impossibly depends on only one feature but multiple features. However, not every feature is related to emotion. Due to this, feature selection and impact were discussed intensively.

2.3.1 Audio Feature Selection

In the age of traditional machine learning methods widely used, human-engineered audio features were very popular to be utilized as model inputs. Previous research has explored plenty of audio features. But it does not mean that combining these features as many as possible could lead to further increases in emotion prediction accuracy, especially when these features do not provide any new emotion-relevant information or even bring disturbance during machine learning.

Facing a large amount of human-engineered audio features, how to evaluate the performance of each feature or feature combination gains much attention. Considering time cost and labour effort, it is impractical to evaluate features one by one manually, especially when the magnitude of features is more than 100. Due to this, some algorithms are applied to identify good features efficiently.

Generally, the three main strategies to select effective features are filter, wrapper and dimension reduction (Zhang et al., 2017). The filter methods use statistical techniques to evaluate the relationship between each input feature and the target response. It is independent of any machine learning algorithm and focuses on

the scores of statistical measures. The commonly used filter methods are ReliefF (Robnik-Šikonja and Kononenko, 2003), mutual information (information gain) and Correlation-based Feature Selection (CFS) (Hall, 2000). Yang et al. (2008) indicated that ReliefF takes the correlation between features into account. Therefore, it is considered a better filter method. According to the data type of features and response (either continuous or categorical), various evaluation metrics are used either for regression problems or for classification problems, such as Pearson's correlation coefficient, Analysis of Variance (ANOVA) and Chi-Squared test. The wrapper methods use a predictive model to find out well-performing subsets of features, so it is beneficial to explore feature combinations. However, it is a class of model-dependent methods usually with high computational costs. The common wrapper methods include Forward Selection (FS), Backward Elimination (BE) and bi-directional elimination (stepwise selection). Dimension reduction methods are also mentioned as embedded methods that project all features into a lower-dimensional feature space. This class of methods usually achieves feature selection in their intrinsic model process. For the classification problem, the decision tree algorithm is a typical embedded method. In regression analysis, Least Absolute Shrinkage and Selection Operator (LASSO) with L1 penalty (Tibshirani, 2011) and Ridge Regression with L2 penalty (Hoerl and Kennard, 1970) are often implemented to shrink many features to zero or almost zero. Such built-in regularization could reduce overfitting effectively.

These feature selection methods are widely used in the MER research area. Yang et al. (2008) utilized regression ReliefF to rank top-m selected features by importance for valence and arousal from a set of 114 features. Panda et al. (2018) extracted 1,702 features and filtered out features with lower weights measured by ReliefF. Saari et al. (2011) argued that wrapper methods could improve the classification performance in music emotion recognition considering the generalizability and simplicity of training models. Zhang et al. (2017) collected eight features to research the feature impact on the arousal dimension. Based on this, they compared various feature selection methods and concluded that the shrinkage methods (belonging to embedded methods) outperform wrapper methods and are similar to entropy-based

filter methods. Through these feature selection approaches, audio feature impact on emotion could be identified.

In recent years, deep neural network models have shown their powerful capability of extracting features autonomously. Due to this, some low-level features or raw audio signals could be taken into account as model inputs without much human intervention. Among these features, time-frequency audio representations such as spectrogram or mel spectrogram are most commonly used for MER (Bian et al., 2019; Nayal et al., 2019). Moreover, there is another increasing trend toward using audio signals as training model inputs directly for reducing model complexity and work effort on audio pre-processing. Orjesek et al. (2019) demonstrated that a model using raw audio samples could outperform a model using pre-processed audio features. In my thesis, the MER experiments are conducted in two scenarios, mainly focusing on audio sources. In one scenario, raw audio clips are fed into stacked deep neural networks to detect the emotional variation. In another scenario, the segment-level log-mel spectrogram is adopted as model inputs and final emotion categories are predicted through a deep learning framework.

2.3.2 Audio Feature Impact

Based on the valence-arousal (VA) emotion definition model, much research has explored dedicated features for emotion detection. Gabrielsson and Lindström (2001) pointed out that valence is related to mode (major/minor) and harmony (consonant/dissonant) while arousal is related to the tempo (fast/slow), pitch (high/low), loudness (high/low) and timbre (bright/soft). Yang et al. (2008) concluded that spectral shape and pitch are the top features related to arousal. In contrast, energy-related features are not much relevant to arousal, and the top features more related to valence are rhythmic (beat and tempo) and pitch properties of sound. Further, Yang and Chen (2011) used MIR Toolbox to extract more features for each dimension of the VA emotion model. They classified the features into three sets: melody, timbre and rhythm, and argued that these features are related to emotion perception more closely. Panda et al. (2018) used selected standard features plus novel features to identify which features have better effects on each dimensional quadrant.

As shown in their experiments, except novel features they proposed, tone colour features (related to timbre) contribute more to emotion recognition. Grekow (2017) mainly used Essentia and Marsyas tools to extract music features and found that low-level features (a full list of low-level features, please see the website [¶]) take main effect for the detection of both valence and arousal. Apart from that, rhythm features impact arousal detection more, while tonal features which describe keys and chords are more beneficial for valence. This research shows that some audio features were commonly selected and showed the importance of emotion recognition, such as rhythm and timbre. However, there are no unified feature sets to be applied. To a great extent, this depends on what kind of music feature analysis tool researchers used. Even for the same feature or feature categories, different tools provide different subdivisions of acoustic attributes. Moreover, a feature might reflect different degrees of influence on arousal and valence. This could explain why researchers set up different experimental environments regarding music datasets, feature selection methods, learning models, evaluation metrics, etc.

2.4 Dataset Introduction

This section introduces some well-known datasets used in MER tasks. Generally, these datasets include music excerpts with emotional annotation. Some contain lyrics, song metadata, acoustic features, tags, etc. Table 2.4 lists the details of these datasets.

Most datasets provide music clips with a duration of 15-60 seconds in MPEG layer 3 (MP3) format. Compared with no audio provided, these datasets are more flexible for extracting features through deep neural networks. Moreover, the data scale is an important factor that affects learning performance. Large-scale data could be more likely to reduce overfitting. Contrasting to image and NLP research that used thousands to millions of training samples in deep learning, the data scale for MER is relatively small (mainly ranging from hundreds to thousands) due to the unavailability of annotation. This might make it more challenging to conclude

[¶]https://essentia.upf.edu/streaming_extractor_music.html

from deep learning experiments. Although lacking the quality to some extent, I attempted to apply neural network models to MER datasets to improve the MER performance. Otherwise, data augmentation technology might be utilized to increase data diversity. Regarding data quality, annotation consistency should be guaranteed. Due to emotional ambiguity and context changes, measuring emotion precisely is difficult. Generally, dimensional annotation is more practical for both classification and regression problems. Further, a good dataset must consider balanced samples, especially in terms of emotion distribution and even genre, which would benefit model training. From Table 2.4, we can see that the data scale of MSD and Music4all is large, but they lacked either audio resources or the annotations my experiments needed. Therefore, the datasets with required samples as many as possible are selected, such as PMEmo, AllMusic and emoMusic.

Table 2.4 : Music Emotion Dataset Introduction

Dataset Name	**Scale**	**Annotation**	**Audio Source**	**More Information**
DEAM	1,802 excerpts	valence, arousal	45-second music clips in MP3 format	(Aljanaki et al., 2017)
DEAP	120 excerpts	valence, arousal, dominance	1-minute video	(Koelstra et al., 2012)
emoMusic	744 excerpts	valence, arousal	45-second music clips in MP3 format	(Soleymani et al., 2013)
emotify	400 excerpts	9 emotional categories	1-minute clips in MP3 format in 4 genres (rock, classical, pop, electronic)	(Aljanaki et al., 2016)
GMD	1,400 songs	genre, valence, arousal	Greek songs, no audio but YouTube links provided	(Makris et al., 2015)
CAL500	500 songs	a set of 174 different tags	no audio available	(Turnbull et al., 2008)
MoodSwings	240 excerpts	valence, arousal	30-second music clips, no audio available	(Kim et al., 2008)

Dataset Name	**Scale**	**Annotation**	**Audio Source**	**More Information**
AMG1608	1,608 excerpts	valence, arousal	30-second music clips, no audio available	(Chen et al., 2015)
Million Song Dataset (MSD)	1,000,000 songs	linked with Last.FM corpus	no audio available but song metadata provided	(Bertin-Mahieux et al., 2011)
PMEmo	794 excerpts	valence, arousal, EDA signals	the chorus part of pop songs in MP3 format	(Zhang et al., 2018)
Music4all	109,269 excerpts	valence, energy, genres, tags and so on	45-second music clips in MP3 format	(Pegoraro Santana et al., 2020)
soundtracks	360+110 excerpts	Categorical (tension, anger, fear, happy, sad, tender) and Dimensional (valence, energy, tension)	15 seconds clips in MP3 format	(Eerola and Vuoskoski, 2011)
AllMusic	1,000 excerpts	4 quadrants in VA space	30 seconds clips in MP3 format	(Panda et al., 2018)

Chapter 3

Literature Review for Music Emotion Recognition and Social Tags Analysis

This chapter introduces related work about music emotion recognition and social tags analysis. It includes emotion response time to music and music segmentation, traditional machine learning methods for MER, neural network methods for MER, multimodal with fusion strategies, social tags applications and cross-domain methods.

3.1 Emotion Response Time to Music and Music Segmentation

Similar to feature selection, it does not mean that the longer music stimuli could convey more effective cues for emotion recognition. That's why MER research usually analyzed music excerpts rather than the whole songs as experimental objects. Bigand et al. (2005) studied emotion response time to music and the emotion perception from music excerpts with different duration. Through groups of experiments, they demonstrated that music of 1-second length contains enough cues to evoke emotion in the listeners, especially when these listeners previously experienced these cues. Xiao et al. (2008) thought music emotion may vary within each song and then discussed what is the best segment duration to present stable emotion. They tested 4 versions of datasets with different duration of music excerpts ranging from 4 seconds to 32 seconds. The results show that the better duration for emotion recognition is 8 seconds and 16 seconds while 32 seconds might not be good due to emotion changing over a relatively long time. Nordström and Laukka (2019) tested the response time for different emotions such as anger, happiness and sadness. They concluded that stable emotion could be recognized within a million-second level (250ms). The

longer duration may lead to emotional change. So it is acceptable to use such short intervals to recognize music emotion.

Based on the research on response time, researchers generally conducted two types of emotion recognition: static and dynamic (Yang and Chen, 2012). Static emotion recognition usually provides an emotion label or value to represent a song or a music excerpt. Referring to Table 2.4, the length of the music pieces for static annotation usually ranges from 15 seconds to 45 seconds. While dynamic emotion recognition focuses on tracking music emotion variation. In this situation, each short time interval of the song is given an annotation. Those labels are usually numeric data. In practice, such intervals could be 0.5 seconds or 1 second, matching annotation sampling frequency of 2 Hz or 1 Hz.

To promote the MER performance, the appropriate length and excerpted position for songs are chosen to balance acoustic homogeneity and surrounding context. Zhang et al. (2018) collected the 30-second chorus part for each popular song to achieve higher annotation consistency. Wu et al. (2014) argued that song-level features may lead to inaccurate feature representation for emotion recognition due to music emotion varying between segments. However, emotion is mostly consistent within each segment. Further, Aljanaki et al. (2015a) distinguished emotional segments from structural segments for music. They compared these two types of segmentation and found that emotional boundaries coincide with structural boundaries very often. Therefore, segment-level emotion detection for music is reasonable. In practice, researchers kept the original annotation for longer audio inputs but carried on segment-level analysis and recognition through machine learning models. Lee et al. (2018) compared sample-level deep learning with frame-level one by configuring convolution filter length and stride rather than partitioning the raw audio at first. The segmentation occurs during training, which leads to no way to obtain segment-level data for additional manipulation. In contrast, Sarkar et al. (2020) indeed divided each audio clip into 5-second segments and transformed them into mel spectrogram as inputs to VGGNet-style model (Simonyan and Zisserman, 2015). But they assigned original labels to segments as their training targets and

set up rules to make the final decisions, which may mislead the final prediction. The problem is there is no good way to avoid using segment-level annotation.

In my research, dynamic and static emotion recognition for music is studied separately. For dynamic emotion recognition, 0.5 seconds is used as the sample duration to detect emotion variation in VA space. For static emotion recognition, we feed time-series segments of each music clip to the training model without requiring new segment-level emotion labels and predict the final emotion classification.

3.2 Music Emotion Recognition Methods

Generally, music emotion recognition involves analysis for three types of problems: classification, regression and clustering. For the classification problem, the predicted targets are one finite set of discrete emotion categories. In comparison, the regression analysis aims to estimate the relationship between input features and continuous values. Clustering analysis usually groups songs with social tags according to their similarity, especially when emotion annotation is unavailable.

Basically, there are two learning paradigms for MER: supervised learning and unsupervised learning, depending on the target labels given or not. Supervised learning trains a model to determine the pattern between the inputs and the labelled outputs. Unsupervised learning analyses unlabelled datasets to find out underlying data correlation and representation automatically (Dieleman et al., 2011). Additionally, falling between supervised learning and unsupervised learning, semi-supervised learning combines a small amount of labelled data with a large amount of unlabelled data during training (Wu et al., 2013). To solve classification or regression problems, we often mention supervised learning (Laurier et al., 2010; Chung and Yoon, 2012). While clustering is achieved by unsupervised learning (Laurier et al., 2009).

3.2.1 Traditional Machine Learning Methods for MER

Before the emergence of neural networks, traditional machine learning algorithms were applied extensively in the MER research area. These models usually require plenty of pre-processed human-engineered features as model inputs to build up learn-

ing patterns. The data type of emotional response (either discrete categories or continuous values) determines whether the learning models serve a classification problem or a regression problem. Here some of the most popular learning methods and their applications in MER are introduced.

The classification-based models mainly include Support Vector Machine (SVM), Decision Tree, K-Nearest Neighbors (KNN), Naive Bayes and Logistic Regression.

SVM (Cortes and Vapnik, 1995) is a supervised machine learning algorithm that can classify data into two classes by finding hyperplane in n-dimensional feature space. The best hyperplane is the one with the maximized distances between the nearest data points and the hyperplane. SVM is effective in high-dimensional feature spaces but is not suitable for large-scale datasets or datasets with much noise.

A decision tree uses a tree-like structure. Each non-leaf node represents one decision based on each feature. Data is classified along "branch" level by level until 'leaves' where a class or a probability distribution over the classes is identified. Decision tree models are easy to interpret and perform well for large datasets regarding computing resources. But such models are prone to overfitting due to much dependence on the training data. The Random Forest algorithm is an extension of the decision tree, where you first construct multiple decision trees with training data, then fit your new data within one of the trees. Superior to a decision tree, it could avoid sorting data into one irrelevant category.

KNN is a non-parametric classification method that uses the training dataset to find the k closest relatives. It is one of the most simple machine learning models. The k value and distance function are the main hyperparameters to impact the final prediction. If the data scale is large, the KNN model is not a good choice on account of computation cost.

Naive Bayes is based on Bayes' Theorem, which calculates the probability of whether a data point belongs within a certain category or not. Because only probabilities are outputs, it runs fast and works well on multi-class prediction. But the prerequisite for Naive Bayes models is the assumption of strong independence among

features. This may limit the accuracy of prediction to some extent since many music features are not orthogonal.

Logistic Regression uses a logistic function to calculate the probability of the output so as to predict a binary outcome. It is also a fast and simple algorithm. But different from Naive Bayes, which optimizes joint likelihood as a generative model, Logistic Regression trains a discriminative model by optimizing the conditional probability based on inputs. So it can not be applied to non-linear classification problems and is sensitive to outliers.

Researchers compared various classification-based learning methods to find the better one for the MER tasks. Laurier et al. (2010) adopted nine algorithms, including four different SVM algorithms, two different decision tree algorithms, K-NN, logistic regression and Gaussian Mixture Models (GMMs) as training models and find which algorithm can predict a certain emotion category with the highest accuracy. As a result, SVM showed the best performance. Kartikay et al. (2016) applied four algorithms - SVM, Naive Bayes, decision tree and Linear Discriminant Analysis (LDA) to the same dataset to compare the accuracy and find that LDA could generate the best result. Sharma et al. (2020) conducted a comparative study between linear SVM, decision tree, Kernel SVM, K-NN, Naive Bayes, logistic regression and random forest to classify high/low valence and arousal based on audio features. The conclusion is that SVM has the maximum accuracy for both dimensions. Actually, SVM is the most popularly used for emotion classification (Han et al., 2010; Lin et al., 2010; Panda et al., 2018).

For regression-based models, linear regression is a widely used approach for modelling the relationship between one or more feature inputs and a continuous output. It is also termed Multiple Linear Regression (MLR) for more than one independent variable. This method is easy to implement and fast to train. But it is only fit to linear relationship and assumes that input features are uncorrelated with each other. Besides linear regression, SVM, decision tree and KNN could also be used in regression models where the outputs are continuous values. In this situation, SVM usually appears as Support Vector Regression (SVR) (Schölkopf et al., 2000) for

both linear and non-linear problems. SVR uses the same principles as the SVM. The decision tree is called a regression tree or regression tree-based random forest for non-linear problems. It splits a data set into smaller subsets resulting in a tree with decision nodes and leaf nodes. KNN regression calculates the mean of k nearest data points as the output. The advantages and disadvantages of SVR, regression tree and KNN are similar to those used in classification.

Yang et al. (2007) formulated emotions as continuous valence and arousal (VA) values and employed MLR, SVR and regression trees (Solomatine and Shrestha, 2004) to test the prediction accuracy. Consequently, SVR showed the best results. Based on the VA emotion plane, regression-based models are usually used to evaluate best music feature sets for MER (Yang et al., 2008; Grekow, 2017; Nawaz et al., 2018; Vatolkin and Nagathil, 2019).

Table 3.1 lists some papers mentioning the typical traditional machine learning methods in MER, but not limited to these papers. Generally, these methods are able to achieve better performance by selecting a set of appropriate features. However, that usually involved large time and labour costs for feature extraction and selection. Together with the inherent limitation of these algorithms, the dataset scale is usually not big, such as 195 songs (Yang et al., 2008), 324 songs (Grekow, 2017). That's the motivation to move forward to deep learning methods.

Table 3.1 : Literature for traditional machine learning methods for MER

Predicted Data Type	**Literature**	**SVM**	**SVR**	**Naive Bayes**	**MLR**	**Decision Tree(s)**	**Regression Tree(s)**	**KNN**	**Logistic Regression**	**LDA**	**GMMs**
Categories	(Laurier et al., 2010)	✓				✓		✓	✓		✓
	(Kartikay et al., 2016)	✓		✓		✓				✓	
	(Sharma et al., 2020)	✓		✓		✓		✓	✓		
Clusters	(Patra et al., 2016)		✓								
	(Panda et al., 2015)	✓		✓				✓			
3D regression	(Deng and Leung, 2015)		✓								
2D	(Yang et al., 2007)		✓		✓		✓				

Predicted Data Type	**Literature**	**SVM**	**SVR**	**Naive Bayes**	**MLR**	**Decision Tree(s)**	**Regression Tree(s)**	**KNN**	**Logistic Regression**	**LDA**	**GMMs**
regression	(Grekow, 2017)		✓				✓				
	(Nawaz et al., 2018)				✓						
	(Vatolkin and Nagathil, 2019)				✓						

3.2.2 Neural Network Methods for MER

In recent years, neural network learning, as one class of machine learning methods, has developed quickly and applied to many research fields, including computer vision, Natural Language Processing (NLP), speech recognition, machine translation and so on. Based on deep neural networks inspired by information processing in biological systems, multiple layers of neural networks are used to progressively extract high-level features from the raw input without too much human intervention.

Here some common neural network models related to music recognition are introduced.

Convolutional Neural Networks (CNNs) are one kind of well-known Artificial Neural Network (ANN) and are widely applied to image recognition. CNN has the powerful capability of learning feature representation automatically. Generally, A CNN model consists of an input layer, one or more hidden layers and an output layer. Further, the hidden layers usually contain three main types of layers: convolutional layer, pooling layer and fully-connected (FC) layer. Among them, the convolutional layer is the core building block of a CNN model, where a dot product is performed between the convolution kernel and the layer's input matrix. To achieve better performance, stacked multiple sets of CNNs are designed to extract features deeply, such as ResNet (Nayal et al., 2019) and DenseNet (Bian et al., 2019). But the huge learning parameters with computation costs should be considered and balanced. Inspired by the success in image detection, CNNs have been employed in various music research (Choi et al., 2016; Senac et al., 2017; Lidy and Schindler, 2016). As for the input of a CNN model, music audio could be fed in two typical ways: one-dimension (1D) raw audio waveform or 2-dimension (2D) time-frequency representation like an image. Corresponding to this, CNN models could be 1D CNN (Lee et al., 2018) or 2D CNN (Sarkar et al., 2020). In my work, these two types of CNN models are used in different scenarios.

Recurrent neural networks (RNNs) are another famous set of neural network models, especially useful for processing sequential data such as speech, music or natural language. It includes a feedback loop where the output from step $n-1$ is fed

back to the network to affect the outcome of step n, and so forth for each subsequent step. Therefore, RNN models are context-sensitive and show dynamic behaviour over time. For relatively long sequential data, traditional RNNs expose the drawback of short-term memory, which may lose some key context information far away from the current step. To overcome such problems, Long Short-term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) and Gated Recurrent Units (GRU) (Cho et al., 2014) networks are designed to set up some gates for selective context retention. They also belong to RNN models but avoid vanishing gradient problems, especially for long-term dependency from a technical perspective. Further, Bidirectional LSTM (BiLSTM) was proposed (Graves and Schmidhuber, 2005). Compared to those RNN algorithms mentioned before that only leverage previous context, BiLSTM can access context forwards and backwards to capture more information.

Since music is time-series data, RNN models are able to capture sequential information, which is a crucial factor in improving the performance of music emotion variation detection. Especially, LSTM and Bidirectional LSTM (BiLSTM) could improve the capability of exploiting contextual information over a long duration and demonstrate superiority in music data processing (Weninger et al., 2014; Coutinho et al., 2015; Li et al., 2016).

Further, combined with the feature learning of CNNs and sequence learning of RNNs, integrating these neural networks have been implemented in music applications to gain better performance. Choi et al. (2017) proposed a convolutional recurrent neural network (CRNN) for music classification. This model showed a stronger performance compared to those models using CNN only in terms of the scale of parameters and training cost. Malik et al. (2017) took advantage of the capabilities of CNNs and bidirectional GRU to detect music emotion based on VA space. This kind of stacked network achieved significantly better results than traditional machine learning models and the RNNs-only model.

Besides CNN and RNN, other neural networks also contributed efforts to MER research. Li et al. (2015) utilized Deep Belief Network (DBN) to extract high-level lyrics features and joint bag-of-character to get better performance. (Bhattarai

and Lee, 2019) fed feature representations from pre-trained CNNs to Multi-layer Perceptron (MLP) to detect music emotion in VA emotion space. For electroencephalogram (EEG) signals from the neurons of the brain, researchers often utilized Graph Neural Networks (GNNs) to deal with data in such graph domain, thereby achieving EEG-based emotion recognition (Zhong et al., 2020).

Generally, neural network models are more effective than traditional machine learning models. The output of a neural network model could be either classes, probability values or real numbers depending on the appropriate loss function and specific activation function in the output layer. So it is more flexible and generalized. Moreover, neural network methods require few human-engineered features so as to save too much effort on prior knowledge learning and music feature preparation. Due to this, researchers could pay more attention to model design rather than feature extraction. In current MER research, the relatively small data scale limits the performance of neural network models, which usually require large data samples to learn better patterns. However, we could still use neural networks to extract feature representation and achieve better results than traditional machine learning methods in some aspects. According to these considerations, my work mainly uses neural network models to recognize music emotion.

3.2.3 Feature Representation

Artificial neural networks actually calculate feature representation. Distinct from human-engineered features extracted from source data before model training, feature representation benefits from the ability of neural networks to extract inherent information and generate vector-based features to represent sources. Then feature representation can be considered as feature input for another learning model. Here my thesis introduces some model design methods which are based on feature representation from neural networks.

One effective method is transfer learning. It trains a learning model for one task with enough data and then transfers the knowledge gained from this source domain to a second task in related domains. In this way, transfer learning could benefit

the tasks with insufficient training data and leverage knowledge from pre-trained models to solve new problems. Fan et al. (2020) utilized a pretrained model VGGish (Hershey et al., 2017) as a feature extractor where the audio data is converted into latent feature vectors as inputs for subsequent training. Bhattarai and Lee (2019) pre-trained 5-layer CNNs on Million Song Dataset (MSD) (Bertin-Mahieux et al., 2011) and then applied this model to EmoMusic dataset (Soleymani et al., 2013) to obtain a set of feature representations from these five CNN layers, followed by a regression model to predict music emotion.

The unsupervised learning structure with embedded neural networks could also provide feature representation. The commonly used methods are Autoencoder (AE) and Restricted Boltzmann Machine (RBM). An autoencoder is an architecture used to learn efficient codings of unlabeled data, which includes two main parts: an encoder that maps the input into the code and a decoder that reconstructs the input from the code. The encoder output could be viewed as feature representations for the input data. Sometimes, AE is also used to correlate and blend the multimodal features into new features that contain more common information (Xianyu et al., 2016). A Restricted Boltzmann Machine (RBM) is a generative stochastic artificial neural network that can learn a probability distribution over its set of inputs. RBM is usually used in multimedia applications to extract feature representations from multiple sources, such as image-text and audio-video (Srivastava and Salakhutdinov, 2014). Regarding MER research, Huang et al. (2016) used Deep Boltzmann Machine (DBM) based on RBM to extract both audio and lyric feature representation for music emotion classification. Further, Zhou et al. (2019) proposed an architecture to combine the advantages of AE and RBM, where AE trains audio data while RBM trains lyric data. Then they concatenated those two types of feature representations as the input of another supervised regression model to predict emotion. MusiCoder (Zhao and Guo, 2021) combined transfer learning and unsupervised learning. They conducted an autoencoder on unlabelled audio data to build a pretrained model that serves other labelled datasets to form feature representation. The autoencoder is adopted to extract audio feature representations in my emotion classification task.

In that situation, unsupervised learning models usually act as feature extractors, followed by supervised learning models for prediction. Although RBM can also generate representation, the algorithm is based on comparing probability distribution. That is meaningless for segments within one song to compare each other. Additionally, an autoencoder architecture allows CNN modules to be added so that we can enhance image-like spectrogram analysis.

3.2.4 Multimodal and Fusion Strategies

In recent years, much research has tended to employ multimodal methodologies based on multiple feature sources or multiple modalities to take advantage of their complementarity. Another similar method is multi-view representation learning, which means learning features from multiple perspectives/views/modalities.

Referring to feature sources, music audio data is the primary consideration. Compared with traditional machine learning, where tens or hundreds of human-engineered features are selected, the typical inputs for deep neural networks are 1-dimensional (1D) raw audio data (Lee et al., 2018) or 2-dimensional (2D) mel spectrogram (Choi et al., 2016) or a mix of both (Wang et al., 2019). Further, some research (de Berardinis et al., 2020) made use of a Music Source Separation (MSS) module Demucs (Défossez et al., 2019) to generate vocals, drums, bass and other sources from the raw waveform and fed them into deep learning models. On the other hand, some attempts have been made by only using lyrics (Li et al., 2015), (Corona and O'Mahony, 2015) or electroencephalogram (EEG) signals (Tripathi et al., 2017). For multiple sources, the combination of audio and lyrics is the most popular solution (Patra et al., 2016; Huang et al., 2016; Jeon et al., 2017; Bhattacharya and Kadambari, 2018; Zhou et al., 2019). In these models, they extracted features from audio and lyrics respectively and compared the performance of unimodal and bi-modal methods. All of the results showed that multimodal solutions could boost performance effectively. With physiological signals applied to multimedia research areas, researchers also combined audio and Electrodermal Activity (EDA) to establish an end-to-end multimodal framework (Yin et al., 2020). Verma and Tiwary (2017) applied SVM, KNN and MLP to pre-processed EEG with video

features for 3D continuous valence-arousal-dominance recognition.

Corresponding to different feature sources, different models are used for learning feature representation or prediction. For example, Delbouys et al. (2018) extracted audio features through 2 stacked CNNs while extracting embedded lyric features through CNN plus LSTM. For EEG data, Lin et al. (2010) sought emotion-specific EEG features and fed them into SVM to classify four emotion categories. More discussion on handling physiological signals data could be found in the latest review (Bălan et al., 2020; Zhang et al., 2020).

Apart from using cross-domain resources, the single-domain resource could also achieve multi-view learning that extracts features from multiple perspectives, like 3D shape detection (Su et al., 2015). By combining meaningful information from different views, more comprehensive representations may be learned to contribute to subsequent recognition. (Wu et al., 2014) explore music from multiple levels (song-segment-sentence) thereby constructing a hierarchical training model. Such models have performed much better than single-view models (Li et al., 2019). In my research, the multi-view architecture extracts feature from raw audio comprehensively.

Along with multimodal structure, fusion strategies need to be discussed that how to merge different modalities. Generally, there are several types of fusion approaches: data-level fusion, feature-level fusion and decision-level fusion (or late fusion) (Lian et al., 2018). Data-level fusion usually happens before model training. Thus it is called input-level fusion or early fusion. This fusion is applicable to raw data or pre-processed data. The challenge is to make multiple data on the same page regarding sampling rate, data dimensionality and unit of measurement, which may bring extra costs for data preparation or limit the performance of the training model. Feature-level fusion is applied during model training, where feature representations are extracted and then merged. It still needs to synchronize data from different modalities. However, with the help of neural networks, the correlations and interactions between modalities could be exploited. Those end-to-end studies are apt to adopt this strategy (Jeon et al., 2017; Yin et al., 2020; de Berardinis et al.,

2020). In contrast with the fusion methods mentioned above, decision-level fusion uses each data source with model training independently, followed by fusion at a decision-making stage. It is especially suitable for the situation where data sources are significantly varied from each other as it is difficult to unify them (Rozgic et al., 2013; Poria et al., 2016). Delbouys et al. (2018) named feature-level fusion as mid-level fusion and compared it with late fusion. Their experiments showed a clear improvement in VA detection based on mid-level fusion. It also indicates that audio and lyrics correlate in the model training stage. Further, some research built up learning models for making fusion decisions. For instance, Fu et al. (2020) proposed a graph attention network (GAT) to make decision-level fusion. In my multi-view architecture, feature-level fusion is adopted since both views output homogeneous data.

3.3 Music Social Tags Analysis

Regarding music social tags, researchers usually mentioned music annotation as their purpose. In MIR research, annotation provides song labels so that training models can use them as target labels. One method for this purpose is subjective testing, which can be conducted by either experts or candidates, annotating songs in categories (Lin et al., 2016), or rating songs within a predefined range in a numeric space (Grekow, 2017; Yang and Chen, 2011). Another common method is to utilize crowdsourcing resources such as MTurk workers (Aljanaki et al., 2017), collaborative games (Law et al., 2007), social tags (Çano and Morisio, 2017b) or web service (Knautz et al., 2011). Among them, social tags are relatively mature resources to be explored and ready to use. However, social tags also contain problems such as polysemy, misspellings, junk words and popularity bias (Lamere, 2008), which must be preprocessed before being labelled.

3.3.1 Word Representation

Social tags analysis could draw on methods in text analysis research to explore latent semantic features. To facilitate text analysis, researchers usually do word representation using a vector-based model. A document or word is represented

as a vector in such a model, where each dimension corresponds to one feature. Then these vectors could be used in language modelling and feature learning in a variety of applications, such as information retrieval (Ganguly et al., 2015), opinion mining (Giatsoglou et al., 2017), question answering (Bordes et al., 2014), named entity recognition (Lample et al., 2016) and syntactic-semantic parsing (Socher et al., 2013).

Generally, the approaches of vector-based representation could be divided into supervised and unsupervised methods (Turian et al., 2010). In supervised methods, one-hot representation in Natural Language Processing (NLP) means a simple word-based vector. It encodes binary vectors with the same length as the vocabulary size. And only one element is '1' in each vector. Therefore it is easy to represent but fails to capture syntactic (structure) and semantic (meaning) relationships in the text context. Another classic representation is Vector Space Model (VSM) (Salton et al., 1975), a document-based vector using term-specific weights rather than binary data as element values. However, these two methods expose the drawback of data sparsity in large-scale text analysis.

In contrast, unsupervised methods show better effectiveness in handling large-size vocabulary and documents, which generate compact vectors with real value in low dimensions. They reduce the vector sparsity effectively and can better measure semantic similarity with other words. These models are commonly known as word embedding. One conventional word embedding method is distributional representation, the essence of which is dimensionality reduction and utilizing matrix factorization strategies. Among a variety of methods, a popular one is Latent Semantic Analysis (LSA) (Deerwester et al., 1990; Evangelopoulos, 2013), which performs Singular Value Decomposition (SVD). This method has been used widely in tag representation (Laurier et al., 2009), and in music emotion modelling (Levy and Sandler, 2007; Saari and Eerola, 2014; Schindler and Knees, 2019).

In emerging word embedding research, neural networks are leveraged to learn low-dimensional word representations rather than first reducing dimensionality directly. One notable technique in Natural Language Processing (NLP) research area

is `word2vec` (Mikolov et al., 2013). Two typical models for this technique are Skip-gram and Continuous Bag-Of-Words (CBOW). Skip-gram aims to predict context words from a given target word. In comparison, the CBOW architecture tries to predict the target word through its surrounding context. `Word2vec` focuses on context information but poorly utilizes global statistical data. Thus, it captures more syntactic regularities but few semantic regularities. Another popular technique is `GloVe` (Pennington et al., 2014), which is a new global log-bilinear regression model combining global matrix factorization like LSA with local context window like `word2vec`. Due to this, it could cover both semantic and syntactic information better and outperform other models in the aspects of word analogy, word similarity and named entity recognition.

3.3.2 Music Social Tags Application

Social tags could be used in many music applications, such as emotion recognition, sentiment analysis and automatic tagging. Social tags, as a crowdsourcing resource, especially for large-scale music datasets, save annotation costs effectively.

Social tags are utilized for music emotion recognition (MER) in several ways. In a categorical model, researchers usually used tags as ground truth data directly (Lin et al., 2011). While in a dimensional model, much research mainly used subjective experiments (Yang et al., 2008) or mapped tags to other existing emotion definition models (Panda et al., 2018). Then how to use tags to construct dimensional annotation is a real need for large-scale music datasets. Because such a dataset is hard to be annotated manually, considering labour and time cost. Saari and Eerola (2014) have researched this area using conventional text analysis methods to generate Affective Circumplex Transformation (ACT), then calculate songs' emotions based on associated tag weights and tag coordinates. In another research (Delbouys et al., 2018), songs are labelled with continuous arousal/valence values based on emotion tags and crowdsourcing word emotional ratings without considering the impact of other kinds of tags and the popularity (weight) of tags. Few research mentions the neural word embedding methods to explore the latent relationship of social tags and then represent tags based on dimensional music emotion space.

3.4 Correlation Domains with Music Emotion

For music emotion-related research, many state-of-the-art methodologies could be inspired by related research areas, such as genre classification, speech recognition, music auto-tagging and text analysis. Based on music classification, genre classification could be used as a reference. In music information retrieval, genre classification has been a relatively mature research area. Lin et al. (2009) demonstrated that genre and emotion are correlated and complementary for describing music content. Further, they built up a two-layer structure where the first layer is a genre-specific classifier as a precursor to reducing the diversity of songs to benefit emotion prediction in the second layer (Lin et al., 2011). Content-based deep neural networks for genre classification (Zhang et al., 2015; Jeong and Lee, 2016; Senac et al., 2017) could be applied to emotion classification. From the audio signal perspective, speech emotion recognition is also a good reference, especially on extracting signal features and proposing end-to-end models (Palaz et al., 2015; Wang et al., 2020). (Alías et al., 2016) reviewed audio feature extraction techniques for speech, music and environmental sounds. Those audio signals have many common characteristics that apply to similar processing methods Purwins et al. (2019). Regarding music emotion annotation, Lin et al. (2011) collected online emotion tags to construct a large-scale emotion ground truth. To some degree, emotion classification could use almost the same learning models for music auto-tagging (Choi et al., 2016; Lee et al., 2017; Wang et al., 2019). Further, the analysis of music social tags and lyrics data could draw on text analysis experiences in NLP research (Lamere, 2008; Hu et al., 2009; Laurier et al., 2009; Saari and Eerola, 2014). In multimedia applications, the development of deep neural networks has been promoted by image recognition (Krizhevsky et al., 2017), video emotion classification (Soleymani et al., 2012) and facial expressions (Soleymani et al., 2016). Corresponding to various media sources, multimodal solutions and methods for extracting feature representation could be applied in MER research (Zhang et al., 2020). Another example is the attention mechanism that did not originate from the music domain, but this technique has benefited music-related research. Zhao and Guo (2021) proposed an

autoencoder model with multiple layers of multi-head attention, which is also known as the transformer architecture inspired by the research of NLP (Devlin et al., 2019) and speech recognition (Liu et al., 2020). However, the complexity of this approach is very high, and the pre-training duration is beyond 800 hours for each dataset. It may not be productive for some MER tasks to train such an attention model in terms of computing cost.

Chapter 4

Deep Learning Regression Model for Dynamic Music Emotion Recognition

4.1 Introduction

In this work, we focus on detecting dynamic music emotion in 2-dimensional valence-arousal space, which turns MER into a regression problem. The dimensional taxonomy is thought of as a better one to reduce ambiguity issues and reflect time-series emotion variation.

Given the music dataset, we need to determine the strategy of feature collection first. Most previous MER research prepared pre-processed audio features for training models, resulting in some unavoidable problems caused by the pre-processing. Before deep learning approaches became widespread, traditional machine learning models were the main ways to solve classification or regression problems. Such models usually require human-engineered audio features as input (Laurier et al., 2010; Schmidt et al., 2010), which require professional-level acoustic domain knowledge as well. In recent years, deep learning models have gained widespread attention in music recognition tasks (Bian et al., 2019; Jeon et al., 2017; Delbouys et al., 2018). They usually utilized the time-frequency representation (such as mel spectrograms) as input. With the help of the automatic learning capabilities of deep neural networks, these solutions simplify the feature engineering work but still rely on the pre-processing of raw audio. Apart from that, extracting temporal and spectral audio features from raw audio through some kinds of clipping, scaling and transformation may lead to information loss in this procedure.

Considering the above issues, we propose using raw audio signal data as training model input directly rather than pre-processed audio features. In this way, it can avoid expending too much effort on prior knowledge learning and feature extraction.

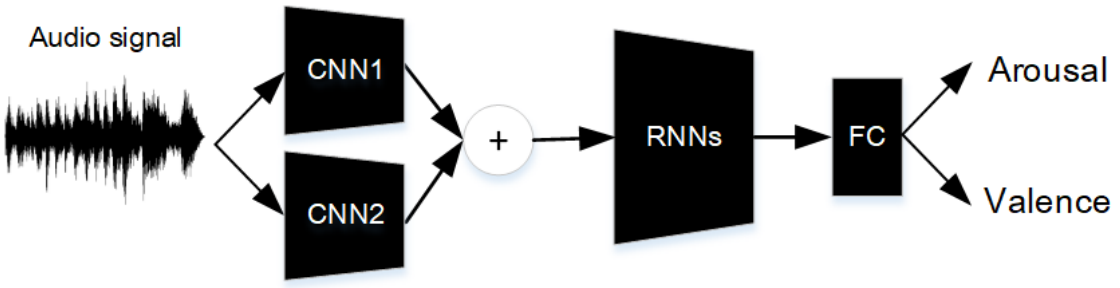


Figure 4.1 : Overview of our multi-view neural networks solution. CNN represents a convolutional neural network, RNNs represent recurrent neural networks, and FC represents a fully connected layer.

Moreover, the raw audio samples with the entire original information are passed to the deep learning model, which makes it possible to learn more comprehensive features with less human intervention.

To process this input, a novel architecture with deep neural networks is proposed for emotion prediction illustrated in Figure 4.1. The multi-view Convolutional Neural Networks (CNNs) are utilized as multiple feature extractors to automatically learn music features from different perspectives. Then these features are aggregated and fed into Recurrent Neural Networks (RNNs) to learn time-varying information for dynamic emotion variation. Finally, the Fully Connected (FC) layer outputs 2 continuous values representing valence and arousal. Based on this structure, the stacked multi-view convolutional recurrent neural network is termed MCRNN. To the best of current knowledge, the proposed model is the first multi-view neural network for music emotion recognition using raw audio signals.

4.2 Methodologies

Facing raw audio signals, we need to design a model that can effectively exploit features from signal data. The architecture of our MCRNN model is illustrated in Figure 4.2. The deep learning model is stacked in feature learning and sequence learning. In the feature learning part, two parallel CNN models are designed to learn features from multiple one-dimension views of the raw audio signal and then

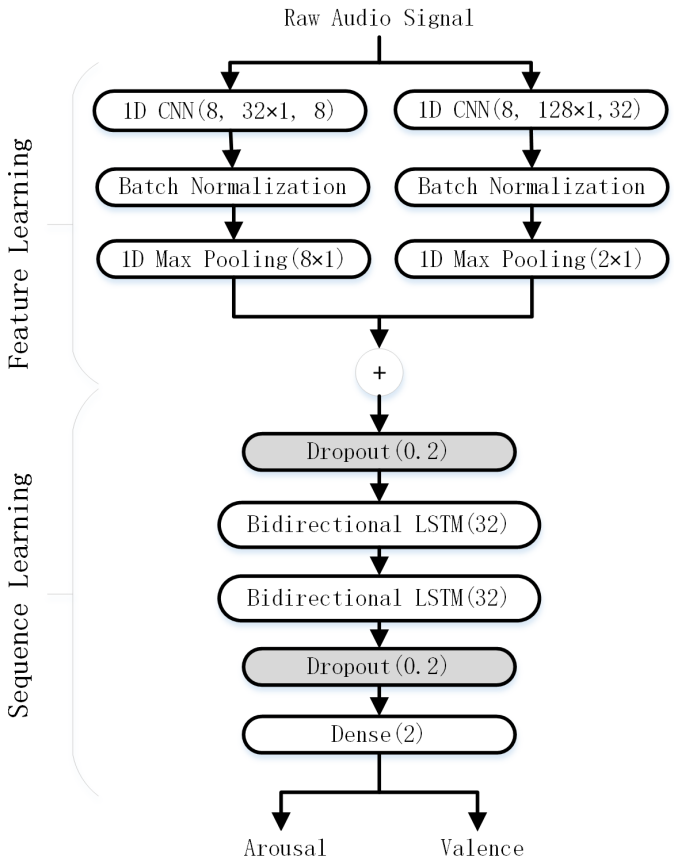


Figure 4.2 : Architecture of the MCRNN model

fuse these features into a single and compact representation. Then, in the sequence learning part, two layers of bidirectional LSTM are employed to learn music contextual information over time. Finally, the output of sequence learning is densely connected into two values representing valence and arousal.

4.2.1 Model Input

As the input of the proposed model, raw sampled audio signals are used instead of traditional engineered features such as MFCC or spectrograms. Here the signal sampling rate is defined as R_s, which means the sampling frequency of audio signal per second. On the other hand, the annotation interval I_a needs to be confirmed in microseconds, during which music emotion is annotated. Based on this, the sequence of audio signals is clipped into non-overlapping training samples. At each time step,

the length of signal sequence as model input L_s is computed by Eqn 4.1.

$$L_s = R_s \times \frac{I_a}{1000} \tag{4.1}$$

For example, given signal sampling rate R_s as 44,100 and annotation interval I_a as 500, then L_s equals 22,050. That is, 22,050 signal samples make up one training model input.

4.2.2 Feature Learning

In the first part of our model, two parallel CNN modules are utilized as multi-view feature extractors. The following parts introduce the algorithm details.

As one important kind of deep learning architecture, CNN was initially designed to analyze 2D data like images. Correspondingly, CNN is applied as 2D CNN. In recent years, along with the increasing demand for dealing with 1D data (like signals) through deep learning, the variant version of 2D CNN has been developed in the form of 1D CNN (Hsieh et al., 2020; Kiranyaz et al., 2021; Zahid et al., 2021). Similar to conventional 2D CNN, the typical structure of 1D CNN contains the input layer, the convolutional layer followed by the pooling layer. The difference is that the input shape of 1D CNN is one dimension rather than two dimensions. That means the algorithm with hyper-parameters needs to be changed, such as kernel size, pooling window size and output shape.

1D Convolutional Layer

Given one model input S, its convolutional feature representation C mainly depends on the weight matrix W, the stride T_c and the number of filters (weight matrices) N. In 1D CNN, the weight matrix is one dimension, and its shape is defined as kernel (filter) size K. The stride means the offset by which the kernel slides to the next analysis window over the data sequence. The number of filters indicates the depth of the convolutional feature map.

Specifically, each unit of the convolutional feature is defined as the Eqn 4.2:

$$c_{n,m} = \sigma\left(\sum_{i=1}^{K} s_{i+T_c\times(m-1)} w_i\right) \tag{4.2}$$

where $c_{n,m}$ is the mth unit of the nth convolutional feature; w_i is the ith weight of the 1D weight matrix; $s_{i+T_c\times m}$ means one unit of input signals, its position is determined by the stride T_c, m and i; $\sigma(\cdot)$ is the activation function. In most situations, ReLU (Rectified Linear Unit) activation function is used. Further, the whole convolutional features could be formulated as below:

$$C_n = \sigma\left(S * W_n\right)(n = 1, 2, ..., N) \tag{4.3}$$

where C_n is the nth convolutional feature; $*$ is the convolutional operation that calculates the dot product of the nth filter W_n with the input signals S; $\sigma(\cdot)$ is still the activation function.

To apply filters across the whole signal sequence and let each signal at the centre of the filters, it is necessary to use "padding" to explore complete convolution features from input signals. The padding means adding zero value to the border of the real sequence data. In this way, filters could cover the whole input sequence. Meanwhile, the padding operation does not affect the convolution result.

In the situation of padding applied, the length of each convolutional feature L_c could be computed as below:

$$L_c = \lceil \frac{L_s}{T_c} \rceil \tag{4.4}$$

where L_s is the length of the signal sequence as one model input mentioned in Section 4.2.1. T_c is the stride. $\lceil\cdot\rceil$ function means taking the least integer greater than an integral part of the division.

1D Max Pooling Layer

Following the 1D convolutional layer, a 1D max pooling layer is applied. The pooling layer is the process of down-sampling feature maps to extract the main features and exclude disturbances. Two common pooling methods are average pooling

and max pooling, which average values and grab the maximum value in the pooling window respectively. The pooling operation usually keeps the same number of represented features as the convolutional layer. For the 1D max pooling layer, each unit of the output from this layer is defined as below:

$$p_{n,m} = \max_{i=1}^{P} \left(c_{n,i+T_p \times (m-1)} \right) \tag{4.5}$$

where $p_{n,m}$ is the mth unit of the nth down-sampled feature; P is the size of the pooling window; T_p is the stride by which the pooling window slides; the max pooling function $\max(\cdot)$ takes the maximum value from a section of nth convolutional feature c_n covered by the pooling window.

In the pooling layer, we still could use "padding" to evenly down-sampling data. Similar to the padding operation in the convolutional layer, the length of each pooled feature L_p could be computed as below:

$$L_p = \lceil \frac{L_c}{T_p} \rceil \tag{4.6}$$

where L_c is the length of each feature in the 1D convolutional layer; T_p is the stride of max pooling; $\lceil \cdot \rceil$ function is the ceiling function as same as the one in Eqn 4.4.

Multi-view CNN Model

In my work, two parallel CNN modules are designed as fine-view CNN and coarse-view CNN separately. Figure 4.3 shows the transformation process of signal data sequence through multi-view layers. To adapt the audio signal input, both CNN modules apply 1D convolutional layers, which receive the same sample sequence. The main difference between convolutional neural networks is the kernel size and the stride. Due to this, the output shapes of convolutional features are usually different. However, both views are kept to have the same depth of feature maps all the time. Further, the outputs of both convolutional layers are normalized for speeding up learning and then go forward to respective max pooling layers. The size of the pooling window in each view needs to be tuned to guarantee the same output shape from two CNN modules so that data from two views can be merged into one set of features for subsequent training layers.

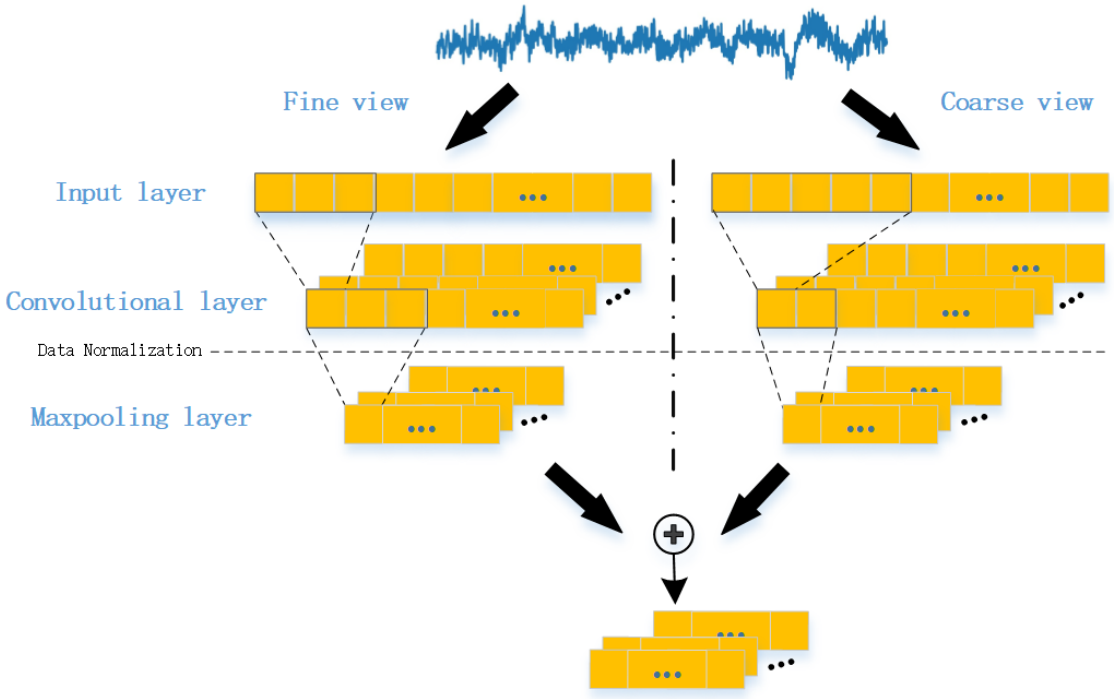


Figure 4.3 : Data transformation through the layers of multi-view feature learning

These two views are similar to sample-level learning and frame-level learning mentioned in (Lee et al., 2017). Sample-level learning uses a relatively small kernel size to detect phase variations within a frame. In contrast, frame-level learning uses a relatively long sample length to capture all possible audio patterns in periodic waveforms. Based on this point, two views are appropriate for learning feature representations for raw audio input. Inspired by this, two filters are applied in parallel to the given inputs in our model. Compared with stacked layers of CNNs that decompose the inputs hierarchically fit for multi-dimensional data, the parallel single-layer CNNs are more proper for learning multiple features from 1D raw audio signals.

As shown in Figure 4.2, the hyperparameter details are given. These hyperparameters were picked mainly based on experimenting with various values and classics from mainstream research. For fine-view CNN, the convolutional layer is configured with kernel size 32×1 and the stride of 8, where the rectified linear unit (ReLU) activation and L2 regularization are applied. The output of the 1D convolutional layer

has the feature maps with a depth of 8, further handled by the BatchNormalization (Ioffe and Szegedy, 2015) layer and one-dimensional max pooling (MaxPooling1D) layer with 8×1 window size to avoid the overfitting issue. Meanwhile, Conv1D in coarse-view CNN adopts 128×1 kernel size and the stride of 32, as well as applies ReLU activation. The following processing is similar to fine-view CNN except changing the pooling window size to 2×1 in the MaxPooling1D layer.

4.2.3 Sequence Learning

The second part of our model is the sequence learning part. Music is a kind of time series data. The recurrent neural network structure is appropriate to explore the pattern from such sequential data. Here is the technical detail about this learning model.

LSTM

Given the data sequence $X = (x_1, x_2, ..., x_T)$, each unit in this sequence is a N-dimensional feature vector that is fed into an LSTM cell. These LSTM cells are connected in order with adjacent hidden states and cell memory passed. In each LSTM cell, there are 3 gates: the input gate i, the forget gate f and the output gate o. These gates are used to determine which features could be forwarded to the subsequent LSTM cells. Figure 4.4 illustrates how the LSTM cells are connected corresponding to sequential inputs and the internal structure of one LSTM cell.

The following equations explain how the gates work to calculate the output of the hidden layer and cell memory.

$$i_t = \sigma\left(W_{xi}x_t + W_{hi}h_{t-1} + b_i\right) \tag{4.7}$$

$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + b_f\right) \tag{4.8}$$

$$o_t = \sigma\left(W_{xo}x_t + W_{ho}h_{t-1} + b_o\right) \tag{4.9}$$

$$c_t = f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \tag{4.10}$$

$$h_t = o_t \tanh(c_t) \tag{4.11}$$

where $x_t (t = 1, 2, ..., T)$ is the tth input vector obtained from the feature maps of our multi-view CNN model; i, f, o represent three gates respectively for this step; $\sigma(\cdot)$

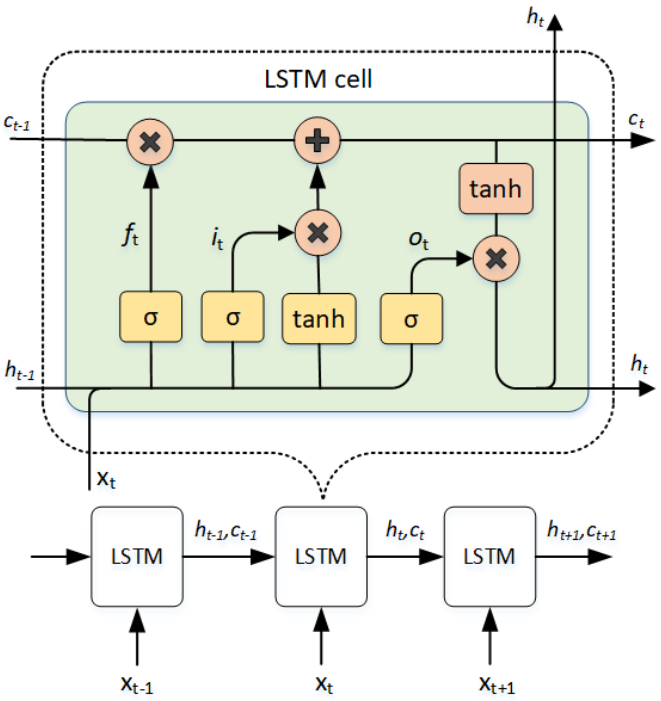


Figure 4.4 : The diagram of LSTM cells sequence and cell structure

denotes the sigmoid activation function; h_{t-1} is the output of the previously hidden layer; b is the bias assigned to each gate, it is usually set as zero by default; W means the weight matrix, each operation node contains particular weight matrices; tanh(·) means hyperbolic tangent which denotes another activation function. Each LSTM cell outputs the cell memory c_t and the output of the hidden layer h_t. For the final output of the LSTM model, either a sequence of hidden states for each time step or the last hidden state is selected, depending on the specific requirement.

BiLSTM

A bidirectional LSTM (BiLSTM) consists of two layers of LSTM models: one LSTM layer in a forward direction and another LSTM layer in a backward direction. Since emotion is associated with the context of music, BiLSTM is a better choice because of its ability to capture both preceding and succeeding information. Besides this, increasing more layers of LSTM neural networks is taken into account. Additional hidden layers can recombine the learned representation from prior layers

and create new representations at high levels of abstraction, and hence disentangle underlying relationships in temporal structure more easily (Pascanu et al., 2014). However, the learning efficiency and training difficulty should be balanced when increasing the size and depth of LSTM models. In this scenario, two bidirectional LSTM modules are employed.

Figure 4.5 indicates how our sequence learning model works. The first BiLSTM module receives the output of our multi-view CNN model. The input for each time step is a N-dimensional feature vector over feature maps. All inputs are fed into the corresponding LSTM cells in the forward and backward layers. The output of the first BiLSTM module is a sequence $y_t^1(t = 1, 2, ..., T)$, each of which is the concatenation of the output of the hidden layer of each LSTM cell in bidirectional LSTM layers. Then, this sequence is fed into the second BiLSTM module with another bidirectional LSTM layer as sequential data input. Unlike the first BiLSTM, we only select the last hidden state y^{2F} from the forward layer and the counterpart y^{2B} from the backward layer and concatenate them as the final output of this module. Finally, the dense layer connects all sequential learning vectors and outputs two regression values representing valence and arousal within a continuous range[-1, 1].

As shown in Figure 4.2, the dimensionality of the output of LSTM layers is set as 32. In the training procedure, the dropout function is added (labelled in grey in Figure 4.2) with the 0.2 rates to further prevent overfitting by ignoring randomly selected neurons, thus reducing the sensitivity to the specific weights of individual neurons. Note that these drop layers are not included in the evaluation stage.

4.2.4 Data Augmentation

Data augmentation (DA) is a method to generate synthetic data to increase the diversity of data for training models. In this way, the model could learn features from more relevant data and reduce overfitting effectively, especially for small-scale datasets. According to music audio characteristics, the commonly used methods include noise injection, time shifting, pitch shifting and time stretch. To keep the same size of model input without changing audio duration and target labels meanwhile

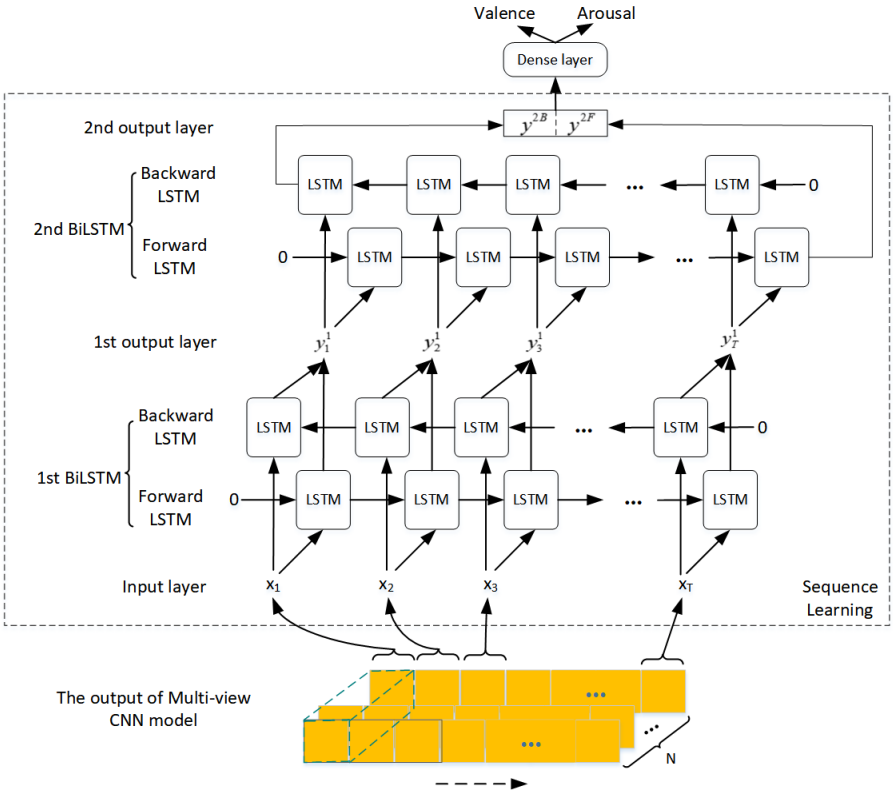


Figure 4.5 : The structure of bidirectional LSTM in our model

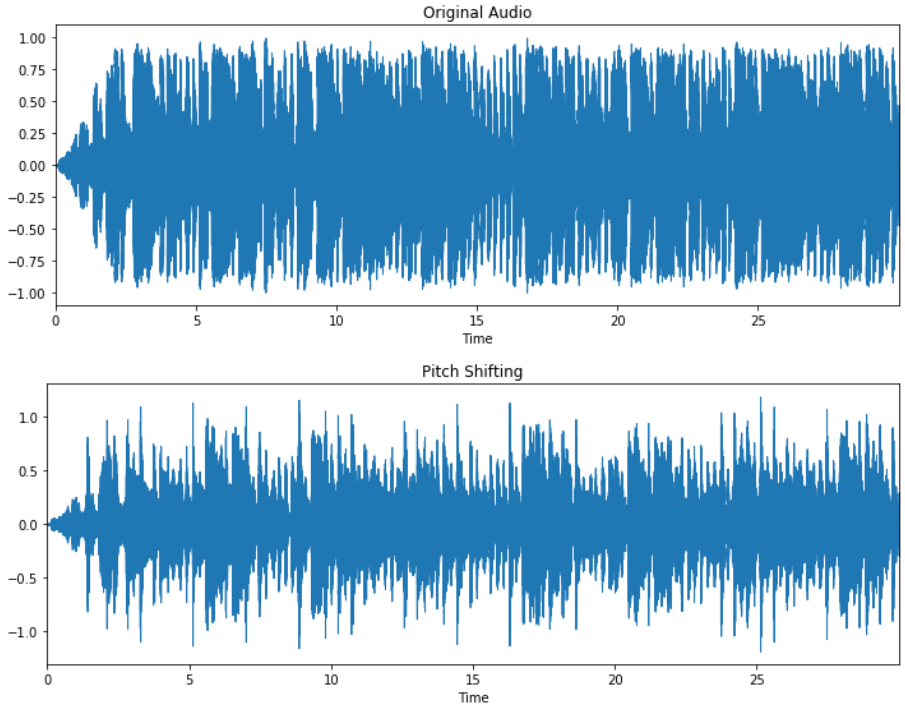


Figure 4.6 : Comparison between original and pitch-shifting audio

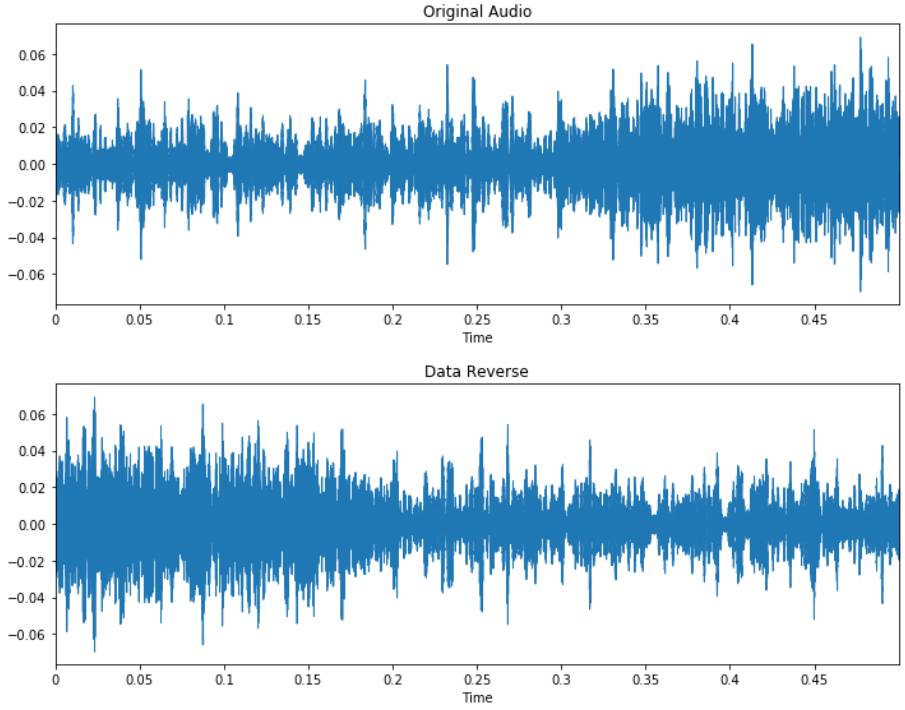


Figure 4.7 : Comparison between original and reversed audio

considering implementation cost, two approaches for audio data augmentation are adopted. One approach is pitch shifting which shifts the pitch of audio clips. Here the pitch of a waveform is lowered by a semitone. Figure 4.6 illustrates the change before and after shifting pitch. Such slight perturbations would increase sample diversity but not impact the original music expression. Distinguishing from other common methods used in audio data, the second approach is data reverse inspired by image processing (Krizhevsky et al., 2017) and time-series application (Wen et al., 2020). This research reverses the raw audio sequences in each annotation interval. Figure 4.7 shows the comparison between original audio and reversed audio during one interval. From the technical view, this reversed data could enhance sequence learning through backward LSTM in our model architecture. From the music perceptual view, the duration of each sample is very short, so the impact on emotional change is negligible. All synthetic data are generated from raw audio samples using `Librosa`* API.

*https://librosa.org/doc/

4.3 Experiments

4.3.1 Data Description

The proposed model is applied to `emoMusic`[†] dataset proposed by Soleymani et al. (2013) which was utilized in MediaEval Emotion in Music Challenge. This dataset is collected from the Free Music Archive (FMA)[‡], including raw audio in mp3 format. Removing a set of duplicates in the initial 1,000 songs, 744 songs are left. For the target labels, songs are annotated via crowd-sourcing on Amazon Mechanical Turk (AMT) in the dimensions of valence and arousal independently, including static ratings given to the whole 45-second clips and dynamic annotations for the last 30-second clips at a frequency of 2Hz. According to the dataset description and related data research (Aljanaki et al., 2017; Vale, 2017), the class imbalance exists in terms of dimensional emotion quadrants, genre and artists. The annotation consistency is also a concern. However, it is the largest dataset available with dynamic valence/arousal annotations and is worthy of study. These limitations should be taken into consideration when analyzing experimental results.

After confirming data information, some preliminary work for model input is carried out. We check the sampling frequency of raw data for each song and keep the songs at 44,100Hz so that our model can be trained with consistent input shape of the raw audio signal. After this filtering, 705 excerpts are retained for our experiments. Then, the last 30-second audio is divided into segments with a duration of 500ms to match the dynamic annotation. According to Eqn 4.1, 22,050 sequential audio samples at each time step are regarded as an input of our model to predict a pair of VA values corresponding to each dynamic annotation.

4.3.2 Evaluation

We evaluate models with 10-fold cross-validation. For each fold, the training/validation/test sets are split with a ratio of 8:1:1. Specifically, all raw audio

[†]http://cvml.unige.ch/databases/emoMusic/

[‡]https://freemusicarchive.org/

files are partitioned into 10 sets with 70 songs in each set. Taking *Fold1* as an example, the first 70 songs are chosen as the test set, followed by the next 70 songs as a validation set. The remainder of the dataset is a training set. Based on this procedure, the dataset is iterated over with the stride size of 70 songs and dividing 3 sets for each fold. It is noticed that the validation set of the last fold is the first 70 songs due to reaching the end of the loop.

Following previous research, Root Mean Square Error (RMSE) is used for model evaluation. RMSE measures the average deviation of the estimates from the observed values, which is an absolute measure of fit. Additionally, R^2 scores are added to compute the coefficient of determination, which is considered a relative measure of fit. Through this approach, the proposed model could be evaluated more comprehensively.

Due to the 10-fold cross-validation implemented, the overall RMSE is calculated based on RMSE metrics in each fold. Given the predicted value $\hat{y}_{ij}$ of the jth test sample in the ith fold and the corresponding true value y_{ij}, then overall RMSE can be defined as Eqn 4.12:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{k} \frac{\sum_{j=1}^{N_i}(y_{ij}-\hat{y}_{ij})^2}{N_i}}{k}} \tag{4.12}$$

where k is the total number of folds, N_i is the total test samples in the ith fold.

And overall R^2 scores are the average of R^2 scores in 10 folds. In the ith fold, given the predicted value $\hat{y}_{ij}$ of the jth test sample and the corresponding true value y_{ij}, the R_i^2 scores of this fold is defined as Eqn 4.13:

$$R_i^2 = 1 - \frac{\sum_{j=1}^{N_i}(y_{ij} - \hat{y}_{ij})^2}{\sum_{j=1}^{N_i}(y_{ij} - \bar{y}_i)^2} \tag{4.13}$$

where N_i is the total test samples in the ith fold, $\bar{y}_i = \frac{\sum_{j=1}^{N_i} y_{ij}}{N_i}$
So overall R^2 scores are defined as:

$$R^2 = \frac{\sum_{i=1}^{k} R_i^2}{k} \tag{4.14}$$

where k is the total number of folds.

4.3.3 Baseline

We take the Deep Neural Network (DNN) model proposed by Orjesek et al. (2019) as the baseline. The dataset it used was provided for the "Emotion in Music" (EiM) task at MediaEval Campaign (Aljanaki et al., 2015b). `EiM` dataset is the extension of `emoMusic` dataset where the development sets have the same source and data distribution (Aljanaki et al., 2017). We argue that the models applied to these two datasets are comparable. Since the baseline model is the same as our research in the aspect of using raw audio as inputs, this baseline model is reproduced on `emoMusic` database. By using the same dataset and evaluation methods, it could keep two models in the same conditions to compare model architectures more convincingly.

The DNN model structure is shown in Figure 4.8. The raw audio inputs are fed into the 1D convolutional layer with kernel size 220×1 (equivalent to 5ms) and a stride of 110 (equivalent to 2.5ms). Similar to our model, the depth of feature maps is 8. The ReLU activation and Batch Normalization are applied. Following this, the convolutional outputs are fully connected into 16 units in a time-distributed way. Then the dropout function with the 0.25 rate is implemented. After that, a bidirectional Gated Recurrent Unit (BiGRU) network handles the data and feeds the results into a fully-connected layer to output final valence and arousal values. For more experimental details, it could refer to the document (Orjesek et al., 2019).

4.3.4 Implementation Details

Apart from model hyperparameters mentioned in Figure 4.2, L2 regularization is added by setting the factor as 0.0001 to reduce overfitting in the fine-view CNN layer. Due to no pre-trained procedure in our MCRNN model, it is crucial to have a good initialization during training. Here the normal initializer proposed in (He et al., 2015) is adopted instead of the Glorot uniform initializer, which produces better performance. Additionally, the batch size is set as 32, and Adam (Kingma and Ba, 2015) is used as the optimizer with a learning rate of 0.001.

In practice, the model training is conducted on the training set for each fold

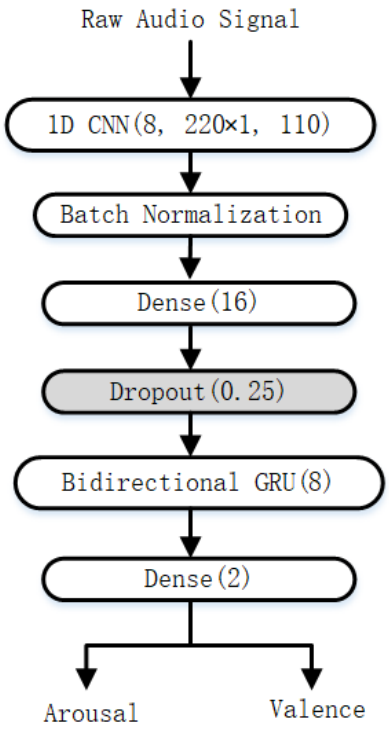


Figure 4.8 : The architecture of DNN model (Orjesek et al., 2019)

with an early stopping strategy. This strategy monitors the loss function of Mean Square Error (MSE) with the patience set to 10 on the validation set. Once the training is finished, the predictive pattern is evaluated on the test set. The training and evaluation are implemented through the Keras library running on top of a TensorFlow backend in Python.

4.4 Results and Discussion

This section compares the proposed model with the state-of-the-art neural network models using either raw audio signals or engineered audio features for music emotion recognition. Then the ablation study is conducted to demonstrate the effectiveness of the solution. Also, the dynamic valence/arousal values predicted by the proposed model and DNN model (Orjesek et al., 2019) are visualized in spatial and temporal views for a model performance discussion.

First, models with different input types are compared based on RMSE metrics

Table 4.1 : RMSE of different neural network models in valence and arousal dimension

Model	**Model Input Type**	**Arousal**	**Valence**	**Average**
DBLSTM (Li et al., 2016)	engineered features	0.225	0.285	0.255
CRNN-NB (Malik et al., 2017)	engineered features	0.231	0.279	0.255
DNN (Orjesek et al., 2019)	raw audio	0.214	0.240	0.227
DNN[a]	raw audio	0.218	0.227	0.223
MCRNN	raw audio	**0.212**	**0.219**	**0.215**

[a]Reproduced DNN (Orjesek et al., 2019) on EmoMusic database

Table 4.2 : R^2 scores compared with the baseline in Valence and Arousal dimension

Model	**Valence**	**Arousal**	**Average**
DNN[a]	0.08	0.405	0.243
MCRNN	**0.133**	**0.430**	**0.282**

[a]Reproduced DNN (Orjesek et al., 2019) on EmoMusic database

in terms of valence, arousal and their average in Table 4.1. The results show that DNN (Orjesek et al., 2019) model using raw audio as input outperforms DBLSTM (Li et al., 2016) and CRNN-NB (Malik et al., 2017) models that use human engineered audio features as inputs. The raw audio inputs contribute to good performance, especially in valence recognition. So we argue that using raw audio signals with appropriate deep neural networks could model features well and gain better performance compared with traditional engineering-feature-based models in this application.

Then, our MCRNN model is compared with the reproduced DNN model in

the `EmoMusic` dataset based on RMSE and R^2 scores. Table 4.1 shows that our model gains lower RMSE scores than the baseline model in both valence and arousal dimensions with an average 4% improvement. Regarding the R^2 scores as shown in Table 4.2, the metric increases approximately 16% on average. Especially in the valence dimension, the result shows a great increment of 66%. Further, to prove the statistical significance of model improvement, the paired t-test is carried out ten folds by measuring RMSE and R^2 scores for these two models. The p-value is less than 0.023 and 0.028 respectively. The standard deviation of RMSE and R^2 scores across folds are 0.02 and 0.1 to give a better understanding of those intra-folds variations.

4.4.1 Performance Results Analysis

Compared with our model, the DNN model only focuses on frame-level feature extraction but ignores phase variation at the sample level. The results show that learning sample-level features could benefit valence recognition more. The outcome confirms that the multi-view architecture could reveal more complementary information from different perspectives, thereby learning more comprehensive features than those the single-view learning solution makes. Valence performance is still lower than arousal when focusing on emotion dimensions, as in most previous research. One possible reason is that audio features might contribute more cues to arousal prediction than valence. Another reason may be the short duration of each music sample, making it difficult to keep consistent valence responses from listeners, thereby impacting the annotation quality. Further, the relationship between model performance and the distributions of music genre and emotion cross folders might be a good point to investigate.

4.4.2 Ablation Study

The ablation study evaluates the effect of multi-view structure and data augmentation. The results are illustrated in Fig. 4.9. Based on R^2 and RMSE scores in valence, arousal and their average, it can be seen that the multi-view CNNs model outperforms single-view models in the same condition of no data augmentation.

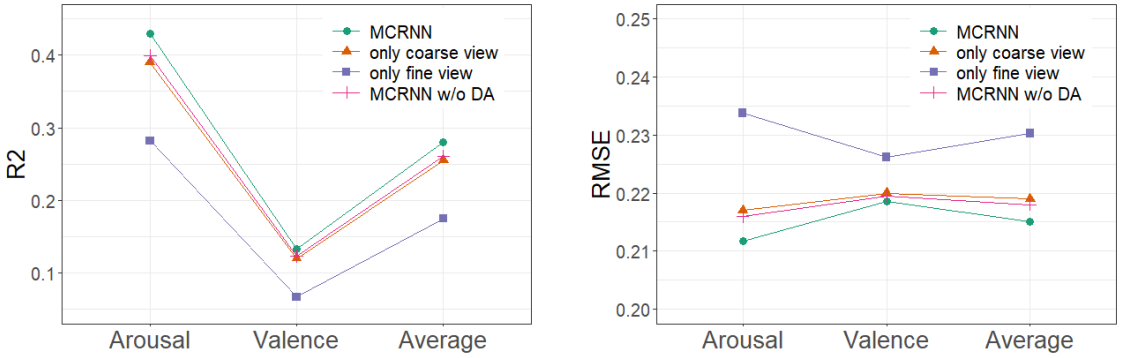


Figure 4.9 : Ablation study based on MCRNN model measured by R^2 and RMSE of valence, arousal and their average. Four situations are compared. That is, our MCRNN solution, single coarse-view CNN, single fine-view CNN and multi-view CNNs without data augmentation.

Further, the coarse-view model shows better performance than the fine-view model. That is, frame-level features play more significant roles in emotion prediction than sample-level features. In addition, the coarse-view model has a slight performance gap against the MCRNN model. It isn't demonstrated that the coarse-view model is enough. Still, there is more potential to improve the collaboration of multiple views and significantly enhance the design of the fine-view model. From the perspective of emotion dimensions, the fine-view model provides more helpful information for valence prediction than for arousal. This could be considered for a new model design in future to train valence and arousal recognition models, respectively, because each emotion dimension may be detected better by different patterns towards different levels (such as sample or frame in raw audio data) of feature representation. On the other hand, data augmentation improves emotion prediction due to increasing the diversity and scale of learning samples.

4.4.3 Performance Visualization

To evaluate the fitness of models intuitively, we visualized the predicted data of our model contrasting with ground truth and reproduced DNN in 2D emotion space and time sequence. One song from *Fold3* and two from *Fold5* are selected.

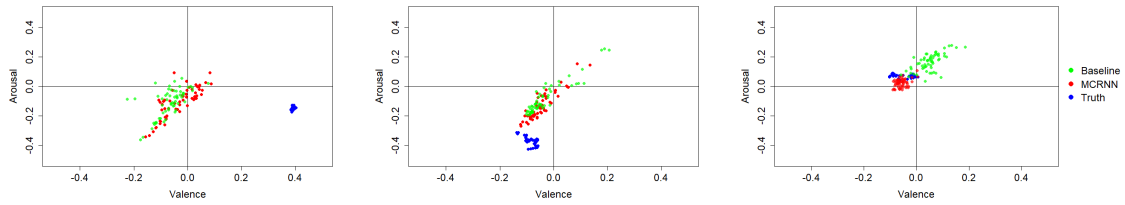


Figure 4.10 : Distributions of dynamic music emotion. Each subfigure represents one song's dimensional Valence (x-axis) and Arousal (y-axis) distribution. Blue data points represent ground truth data; green data points represent baseline predicted data; and red data points represent MCRNN predicted data. 60 data points of dynamic VA values are plotted for each song.

According to the RMSE and R^2 metrics of the 10-fold cross-validation, those two folds show the worst and the best performance. Then the data related to these songs is visualized in 2 views for comparison.

One view is illustrated in Figure 4.10, which visualizes the distribution of points representing time-varying VA values in 2D emotion space regarding ground truth, our MCRNN model and DNN baseline model separately. It indicates that our model prediction is closer to the ground truth than the baseline. Even so, an obvious difference could be observed between the predicted data of the two models and the ground truth. These songs are located in different quadrants based on the ground truth, while both sets of predicted data are distributed close to the origin of 2D coordinates extending to Quadrant1 (Q1) and Quadrant3 (Q3). This may arise from the imbalanced distribution of training data (see Figure 4.11), where most VA values are annotated in the range of [-0.5, 0.5] intensively, and the high proportion of annotations are located in Q1 and Q3. The predicted results tend to fall into these areas to minimise the training loss. For the points deviating from the main distribution, their position would be weakened during training to relieve the loss. For example, in the left subfigure in Figure 4.10, the ground truth is on the edge of data distribution and in Quadrant4 (Q4), which contains the fewest training samples. It can be seen that the prediction is not good. In contrast, the better one occurs in

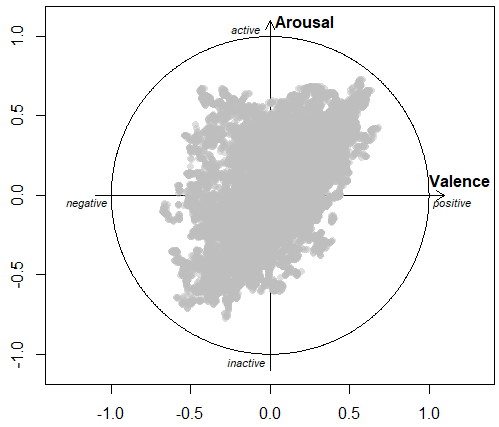


Figure 4.11 : Distributions of dynamic emotion of all songs in 2D space

the right subfigure. Therefore, eliminating data imbalance should be considered as one way to improve model performance.

The other view visualizes the variation of valence and arousal over time in Fi-arousaleffig:timevis. It shows that ground truth is almost a straight line since it is uncommon for listeners to perceive dramatic emotional change within short time intervals. Compared with ground truth, the emotion predicted by either the MCRNN model or the baseline model fluctuate much in a similar waveform. Despite this, both predicted lines go up and down, trending to ground truth. This could be explained by the fact that raw audio as model input leads to this pattern, and at the same time, the target labels lack consistency and precision to some extent. There is still space to optimize model architecture for improving the feature learning from raw audio to final regression output.

4.5 Summary

This chapter introduces novel multi-view neural networks trained end-to-end using raw audio signals directly to predict dynamic music emotion in dimensional valence-arousal space. The experimental results demonstrate that our MCRNN model could perform better than models using pre-processed audio features and

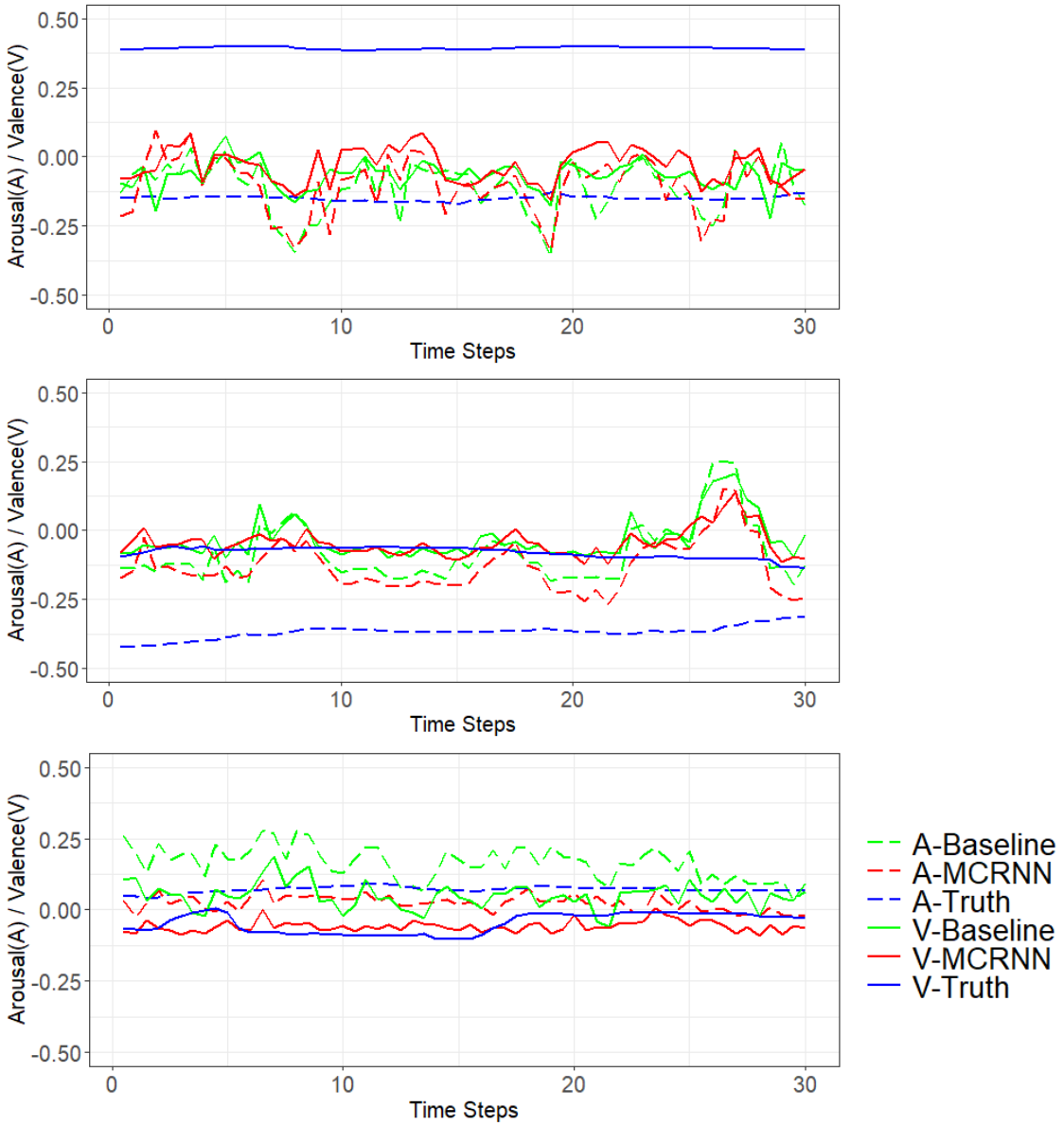


Figure 4.12 : Variation of valence and arousal in time series. Each subfigure represents the variation of valence (solid line) and Arousal (dashed line) over time (x-axis) for one song corresponding to Figure 4.10. Blue lines represent ground truth data; green lines represent baseline predicted data; and red lines represent MCRNN predicted data. Each time step is 0.5 seconds for a total timespan of 30 seconds.

single-view architecture models. In contrast with conventional music recognition methods, our solution does not use crafted audio features, thus avoiding professional acoustic knowledge learning and intense feature engineering effort. Moreover, our model employs multi-view convolutional neural networks stacked by double bidirectional LSTM layers, which could capture more features from multiple perspectives combined with time-series analysis to improve recognition performance. This chapter has resulted in a publication (He and Ferguson, 2020b) , thus validated by peer-reviewers already.

Chapter 5

Deep Learning Architecture for Static Music Emotion Classification

5.1 Introduction

In Music Information Retrieval (MIR) research, emotion recognition is an important branch and benefits various MER application areas. In recent years, deep learning models have become primary methods used to implement emotion prediction (Jeon et al., 2017; He and Ferguson, 2020b). With layers of neural networks, these models are capable of learning music features automatically from raw audio or low-level audio features. In Music Emotion Recognition (MER) tasks, much research is based on music datasets containing emotion annotation, which naturally adopts supervised learning methods to find patterns between each music input and its corresponding annotation. Few studies take into account unsupervised learning for labelled data. In addition, most researchers keep the duration of each audio input in accordance with the given annotation, seldom considering the effect of changing that duration. For dynamic emotion detection, to match the time-varying annotation sampling frequency, which is usually 2 Hz or 1 Hz, the length of each music clip is 0.5s or 1s. These audio clips are fed into a training model and thus implement a one-to-one mapping with those labels (Aljanaki et al., 2017). For static emotion recognition, each music excerpt (usually the duration of 30s or more) corresponds to one annotation. According to this approach, researchers usually extract music features from these music excerpts without further splitting them into shorter segments. However, not all kinds of music duration are appropriate for emotion analysis and model training (Xiao et al., 2008; Yang and Chen, 2012). Some research even splits longer-duration music recordings into a series of short segments but assign presumptive segment-level labels as the training targets rather than using the orig-

inal annotation (Sarkar et al., 2020). Few research has paid attention to adjusting the length of audio input without adding extra annotation.

This study focuses on static emotion recognition and proposes an architecture that uses music segments split from each music excerpt as model inputs while only using the original emotion annotation. Here the framework is divided into two parts. The first part is an unsupervised learning model which generates the feature representation for segment-level music without defining new emotion labels for them. The second part is a supervised training model where segments are viewed as the sequential units of each music excerpt and trained in a deep learning model of handling time-series data to predict the final emotion. In the module of unsupervised learning, the *SpecAugment* technique (Park et al., 2019) is utilized to partially mask log-mel spectrogram input data from frequency and time dimensions to enhance the robustness of the training model.

The main contribution of this work is designing a two-stage MER architecture that combines segment-based unsupervised learning as a feature extractor and supervised learning as an emotion detector. In this way, each music excerpt could be split into contiguous segments without having to provide segment-level annotations. We can feed these segments into appropriate training models to explore potential features effectively. From the perspective of data augmentation, segment-level music with partial masking increases the data scale and data variation for unsupervised learning, thereby boosting the model performance.

5.2 Methodology

A two-stage learning framework is proposed as seen in Fig. 5.1. The first stage is an unsupervised learning model to obtain segment-level feature representation. The second stage is a supervised learning model to predict emotion classification. Regarding feature source, music audio data is used to serve this model structure. For emotion taxonomy, 2D valence-arousal emotion space initiated by Russell (1980) is applied and viewed as a classification problem in this scenario.

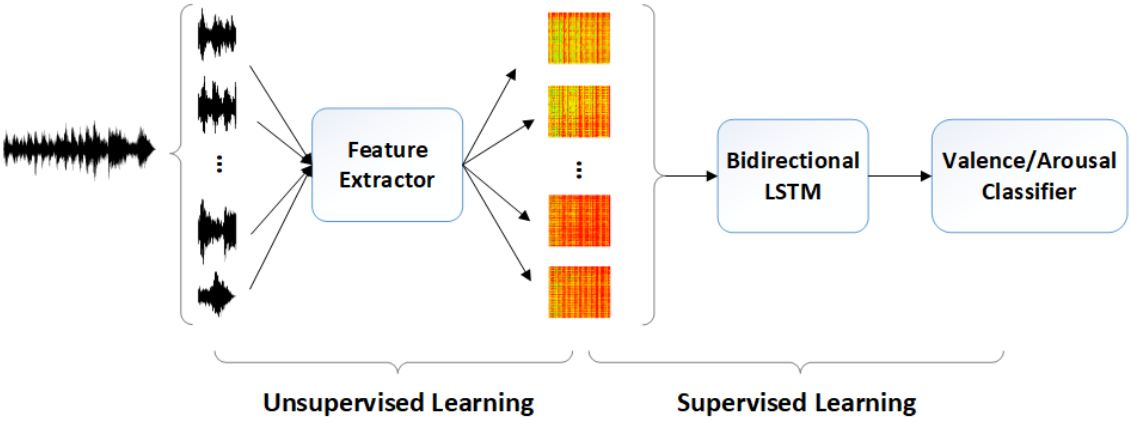


Figure 5.1 : Model overview. The two-stage learning framework includes an unsupervised learning model as a segment-level feature extractor and a supervised learning model as an emotion recognizer

5.2.1 Feature Representation

The detailed design for feature representation is shown in Fig. 5.2. First, each music excerpt is split into segments that are transformed into a log-mel spectrogram. Then the data is partially masked in time and frequency dimensions separately. After that, masked data are passed into an autoencoder architecture to encode and decode to minimise the loss between the reconstructed outputs and the original inputs. In this way, the feature encoder module with the optimized training weights becomes a feature extractor that accepts log-mel spectrogram of segment-level audio data and outputs their feature representation.

Frequency and Time Masking

Inspired by SpecAugment (Park et al., 2019) and MusiCoder (Zhao and Guo, 2021), the input data is partially masked to increase the robustness of the training model against partial loss of information. More importantly, this procedure feeds the model with deliberately perturbed data to reduce overfitting during training. Due to the log-mel spectrogram applied, such data is masked in both the frequency and time domains.

Frequency masking: Given the total number of mel frequency channels F_c,

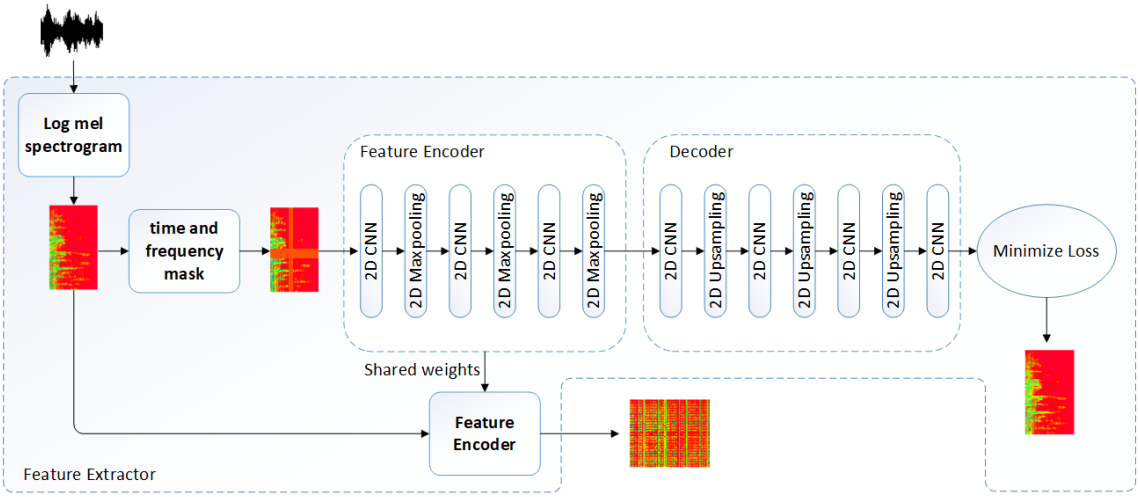


Figure 5.2 : The detailed design for feature representation. Each segment-level audio is transformed into log-mel spectrogram, followed by frequency and time masking. Then such input is fed into a CNN-based autoencoder with the target of minimizing loss. The feature encoder with the optimized weights is used as a feature extractor to provide segment-level feature representations.

the frequency mask parameter F is set and made $F < F_c$. A span of consecutive mel frequency channels $[f_0, f_0 + f)$ is specified to be masked, where f is a randomly selected number from a uniform distribution over $[0, F)$ and f_0 is a randomly selected number from a uniform distribution over $[0, F_c - f)$.

Time masking: Given a log-mel spectrogram with the total time steps T_s, the time mask parameter T is set and made $T < T_s$. A span of consecutive time steps $[t_0, t_0 + t)$ is specified to be masked, where t is the randomly selected number from a uniform distribution over $[0, T)$ and t_0 is the randomly selected number from a uniform distribution over $[0, T_s - t)$.

Here one span of data is masked for each domain respectively. Because the time duration for each segment is not very long and only mel-scaled frequency is included. Masking multiple spans of time or frequency may increase the risk of underfitting during training due to too much information loss. Similarly, the parameters F and T are adjusted based on an appropriate ratio between the width of masking and each

input signal by referring to the previous work (Park et al., 2019; Wang et al., 2020; Chen et al., 2021). For the option of the masked value that replaces the true value, either zero or the mean value could be applied. These two situations are compared in the experiments to find better performance.

Convolutional Autoencoder

Generally, an autoencoder model consists of a feature encoding module ϕ and a decoding module φ. The feature representation $\mathcal{F}$ is the intermediate result of the transition process defined as:

$$\mathcal{F} = \phi(X) \tag{5.1}$$

$$\varphi(\mathcal{F}) \sim X \tag{5.2}$$

where X is model inputs; $\mathcal{F}$ is the output of the encoder; the output of the decoder is an approximation of input X. To achieve this, the autoencoder is trained to minimize the reconstruction error $\mathcal{L}$:

$$\mathcal{L}_{min}(X) = \|X - (\varphi \circ \phi)X\|^2 \tag{5.3}$$

where $\circ$ denotes the composition of function ϕ and φ; squared error is usually used to measure the loss.

As shown in Fig. 5.2, this autoencoder model is a deep CNN-based architecture. The initial design was inspired by deep temporal clustering research (Madiraju et al., 2018) where the encoder outputs latent signal features. Considering 2D spectrogram data as model inputs in my design, multiple layers of CNN are used, which refers to spectrogram-based research proposed by Sarkar et al. (2020). As for the number of CNN modules applied, on the one hand, we need to design enough layers to compress the input data gradually. On the other hand, we need to limit the scale of training weights that would increase with more layers. Since the scale of MER datasets is usually small, using dense layers with too many training parameters regarding overfitting is not appropriate.

Based on such consideration, the feature encoder ϕ consists of 3 groups of stacked layers where each 2D CNN layer is followed by a 2D max-pooling layer. The CNN layers extract latent audio features and the max-pooling layers compact representations. Specifically, given the model input X, it is a 2D matrix representing the segment-level log-mel spectrogram. For 2D CNN, the weight matrix W as the kernel (filter) is a 2D matrix as well. In the CNN layer, each element of the kth convolutional feature map C_k is defined as:

$$C_k[m,n] = \left(X * W^k\right)[m,n] = \sigma\left(\sum_i \sum_j W^k_{i,j} X_{m+i,n+j}\right) \tag{5.4}$$

where each element is the sum of the element-wise product of the input and the kernel (filter); m and n are the indexes of the 2D feature matrix; W^k is the kth filter with i and j as the indexes; $\sigma(\cdot)$ is the activation function. In this equation, the stride is set as one by default and all indexes start from zero.

Following the 2D CNN layer, the 2D max-pooling layer is implemented as:

$$P_k[m,n] = \max_{i,j}\left(C_k[i + T_p \times m, j + T_p \times n]\right) \tag{5.5}$$

where $P_k[m,n]$ is the element of the kth downsampled feature; m and n are the indexes of the 2D feature matrix after the pooling operation; i and j are the indexes of the 2D pooling window; T_p is the stride by which the pooling window slides; the max-pooling function $\max(\cdot)$ takes the maximum value from a bunch of elements in the kth convolutional feature map C_k covered by the pooling window. Still, all indexes in this equation start from zero.

The output of the feature encoder retains the most relevant information of the input and achieves dimensionality reduction. The reconstruction work is implemented by the decoder function φ where a series of 2D CNN layers with 2D upsampling layers are applied. Here the 2D CNN layers perform convolution the same as Eqn 5.4 but increase the depth of feature maps corresponding to the encoder's reverse operation. Cooperating with convolutional layers, the upsampling layers repeat the rows and columns of feature maps to reconstruct the approximations of the original inputs. Similar to the encoder, each CNN layer is applied activation function. In

common situations, the rectified linear units (ReLU) activation function is used to improve training efficiency. Through this unsupervised learning architecture, feature representations are extracted for music segments without labelling the emotion for them.

In practice, the masked log-mel spectrogram data is fed into the feature encoder and trained in the whole autoencoder model. Once the output of the decoder achieves the minimized loss against the original input, the optimized weights are saved for the feature encoder that is used as a feature extractor to generate latent feature representation. The whole processes could be denoted as Eqn 5.6, 5.7, 5.8.

$$\hat{X} = Mask_{F,T}(X) \tag{5.6}$$

$$\mathcal{L}_{min}(X, \hat{X}) = \left\| X - (\phi \circ \varphi)\hat{X} \right\|^2 \tag{5.7}$$

$$\mathcal{F} = \phi_{\mathcal{L}_{min}}(X) \tag{5.8}$$

Loss Function

During the autoencoder model training, the squared error is usually used to monitor the best reconstruction. In this work, Huber loss is used instead. Huber loss is a robust regression loss that is less sensitive to outliers than the squared error loss (Girshick, 2015). This loss function is defined below,

$$\mathcal{L}_{\delta}(x) = \begin{cases} 0.5 \cdot x^2 & if\ |x| \leqslant \delta \\ \delta \cdot |x| - 0.5 \cdot \delta^2 & \text{otherwise} \end{cases} \tag{5.9}$$

where x means the difference between the observed and predicted values. $\delta = 1$ is set by default. In this way, Huber loss could reduce the impact of the outliers and promote training convergence (Zhao and Guo, 2021).

5.2.2 Emotion Classification

The second part of the framework is a supervised learning structure for emotion classification. A Bidirectional Long-Term Memory (BiLSTM) model captures

temporal music information and detects emotion classification. For this model, each input is a sequence of feature representations of time-series segments that constitute one music excerpt. The output is the Valence/Arousal (VA) predictions corresponding to this music excerpt. From the perspective of model implementation, the feature encoder and BiLSTM could be considered as a whole. During training, the encoder module is frozen and holds the optimal weights from unsupervised training while the BiLSTM neural network tunes the weight itself to achieve the final fitting.

5.3 Experiment

5.3.1 Dataset Description

To validate the model, the `PMEmo` dataset* is employed, which is designed for MER research. The dataset contains songs with VA annotations, song metadata, EDA signals, pre-computed audio features, lyrics and even user comments. This music set targets popular songs and collects the chorus part for each song in mp3 format. Among the total 794 songs, my study selects 767 songs that have been labelled with static VA annotations. Regarding annotation consistency, each subject listened to 20 excerpts, including duplicated ones. Each song was annotated by at least 10 subjects, and the bias for repeated annotation from one subject was taken into consideration. So that the quality of the annotation is guaranteed. The chorus excerpts are of various lengths. Most of them are not less than 30 seconds (30s). According to this, the 30s is retained for each song. Song lengths less than the 30s are padded into 30s by repeating themselves from the start to the end. Totally, 230 clips are processed. In this manner, the experiment ensures all music excerpts have the same duration to facilitate subsequent audio processing. More details about this dataset could refer to `PMEmo` document (Zhang et al., 2018). Based on this dataset, the proposed model is compared with previous models to check the effect of audio segmentation and model architecture. However, `PMEmo` dataset has some problems such as single genre (pop music) and imbalanced target labels (see Section 5.3.3). It is necessary to add another dataset to support some viewpoints in the experiment.

*https://github.com/HuiZhangDB/PMEmo

To prove the effectiveness of the model, it is also validated on `AllMusic` dataset (Panda et al., 2018). This dataset contains 900 song clips balanced in terms of Russell's VA quadrants and genres in each quadrant, which avoids the pitfall of `PMEmo` dataset. The quadrantal annotation is obtained based on `AllMusic` emotion tags and Warriner's list (Warriner et al., 2013). A manual blind inspection was conducted to exclude songs with unclear emotions so as to validate the annotation. Most songs are 30-second clips. Only about 2% songs need to be padded to 30s by using the same strategy in `PMEmo` dataset. This dataset is mainly used to check the performance of different segment duration and masking.

Compared with other MER datasets, these two datasets provide raw data with VA annotations, which meet the design requirement. Also, the datasets are the largest ones (refer to Table 2.4) available to use in deep learning.

5.3.2 Audio Processing

This music audio data is processed to prepare the inputs for the training model. First, each 30-second music excerpt is split into contiguous segments. The selection of the segment duration should balance the validity of emotional response and the homogeneity of each segment for feature learning and meanwhile consider the model adaptability. Referring to previous research (Bigand et al., 2005; Xiao et al., 2008; Nordström and Laukka, 2019; Fan et al., 2020), segment duration from the value set of {1s, 3s, 5s, 10s} are tested and the corresponding results are compared. Regarding data normalization, no extra normalisation is required for `PMEmo` dataset due to audio signal values falling into the range $[-1, 1]$. For `AllMusic` dataset, the audio data is normalized into the same range to facilitate the subsequent processing.

Then each segment-level audio is converted into a mel-scaled spectrogram S_m by using the function provided in Python Librosa[†] package. The mel scale is some kind of non-linear transformation of the frequency scale. Its range is analogous to the range of human hearing. The expected data size for each input is 216×128, where 128 represents the number of mel frequency channels while 216 is the number of

[†]https://librosa.github.io/librosa/

fast Fourier transform (FFT) windows calculated from audio data. In order to gain the same data shape for different segment duration to adapt to the model, the size of the FFT window *n_fft* and the number of samples between successive windows *hop_length* need to be adjusted when computing the mel spectrogram. Table 5.1 lists the parameters for mel spectrogram transformation.

Table 5.1 : The parameters for mel spectrogram transformation

Dataset	**Sample Rate**	**Segment Duration**	**n_fft**	**hop_length**
PMEmo	44,100Hz	1s	1,024	205
		3s	1,024	615
		5s	2,048	1,024
		10s	2,048	2,048
AllMusic	22,050Hz	3s	1,024	307
		5s	1,024	512
		10s	2,048	1,024

Note: In this table, 's' denotes second. For `AllMusic` dataset, '1s' segment duration is inapplicable due to the limitation of the model input shape

To reduce the impact of outliers, the value range of S_m is further checked, and then it is transformed into a logarithmic scale base 10. The detail is defined as:

$$S_{lm} = lg(\eta \times S_m + \Delta) \tag{5.10}$$

Based on this, some empirical values of η and Δ are tested, and the value ranges are listed in Table 5.2 for further comparison. The preliminary experiment found that $\eta = 1$ and $\Delta = 1$ could result in a relatively narrow data range with non-negative numbers, which brings about lower reconstruction losses. After converting that, 2D log-mel spectrogram data is transposed to generate the inputs before the masking operation.

For the frequency and time masking, $F = 30$ and $T = 32$ are set. Then the

Table 5.2 : The parameters for log mel spectrogram transformation

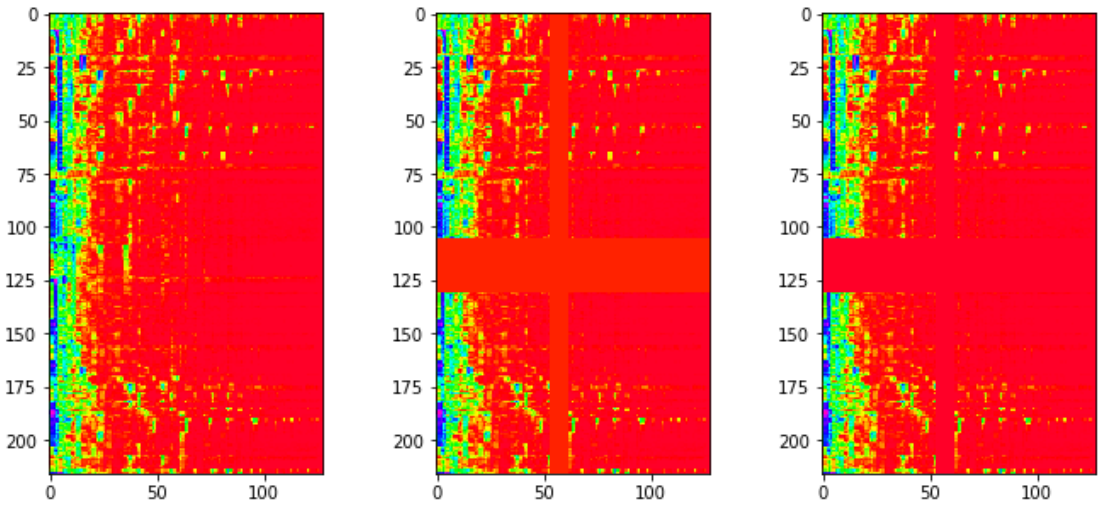
Signal Process Method	η	Δ	**Value Range**	**Mean**
mel spectrogram	-	-	[0, 10,107.66]	12.83
log-mel spectrogram	10	1e-6	[-6, 5.005]	-0.19
log-mel spectrogram	1	1e-6	[-6, 4.005]	-1.12
log-mel spectrogram	10	1	[0, 5.005]	0.76
log-mel spectrogram	1	1	[0, 4.006]	0.33

Note: In this table, we take one set of songs in `PMEmo` dataset as the reference. The segment duration is 5 seconds.

masked spans are padded by either zero or the mean value of the log-mel spectrogram. As observed, the mean value is not zero but the gap is small. Hence, padding the mean value shows a very small increase in performance. Figure 5.3 shows the differences between the original spectrogram, masked spectrogram with mean value and masked spectrogram with zero value. It can be seen that the mean-value span is more distinct than the zero-value span. The following experiments use the mean value to mask the frequency and time spans.

5.3.3 Annotation Transformation

For `PMEmo` dataset, the original annotation data was based on subjective responses in the range from 1 to 9 for both valence and arousal and had been scaled into $[0, 1]$ in the form of continuous values for storage in the dataset. To consider this task a classification problem, these continuous values must be transformed into categories. The distribution of the annotation data is observed in 2D emotion space as seen in Fig. 5.4. Quadrantal classification is not appropriate due to imbalanced training samples in each quadrant (see Table 5.3). Thus, binary classification based on high/low levels for each dimension is adopted. Further, the method used in (Yin et al., 2017, 2020) is drawn on to adjust the neutral threshold. In detail, K-means clustering is applied to generate two clusters, followed by calculating two cluster



(a) (b) (c)

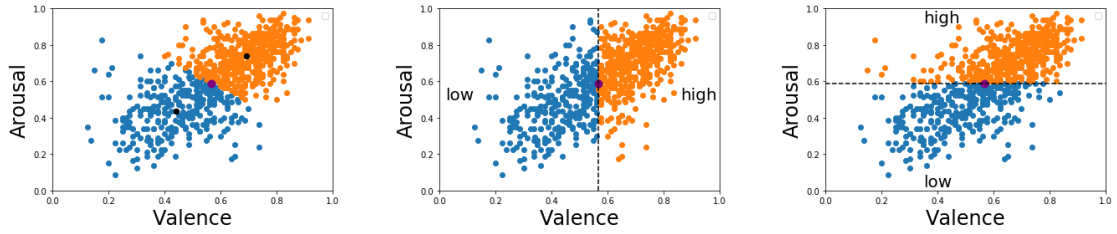
Figure 5.3 : The log-mel spectrogram with masking. (a) the spectrogram without masking (b) spectrogram masked by mean value (c) spectrogram masked by zero

centers and their midpoint. Then the threshold lines are set up for each dimension on the basis of the coordinates of the midpoint. In this way, the labels for training data could be balanced in each category.

In `AllMusic` dataset, the original annotation is balanced quadrants. In accordance with the predictive targets and the annotation used in `PMEmo` dataset, quadrants are transformed into high/low valence and arousal labels.

5.3.4 Training Model Setup

In the unsupervised learning stage, the masked data is fed into a CNN-based autoencoder model. The parameters of the proposed neural networks are given in Table 5.4. All of the 2D CNN layers are specified 3×3 kernel size with one stride. 2×2 pool size with a stride length of 2 is applied for 2D max pooling layers, and the same size is applied for 2D upsampling layers as well. The depth of feature maps starts with 128 and decreases layer by layer in the encoder, then increases correspondingly in the decoder ending with 1 to return to the initial shape. During optimization, the L2 regularizer applies a penalty to the output of the first CNN



(a) (b) (c)

Figure 5.4 : The distribution of static emotion annotation and the division for target classes. (a) 2 centers of k-means clustering and their midpoint (b) binary classification for high/low valence (c) binary classification for high/low arousal

Table 5.3 : The distribution of training samples in each quadrant based on PMEmo dataset

Quadrant	**Training Samples**	**Percentage**
Q1	382	49.8%
Q2	76	9.9%
Q3	224	29.2%
Q4	85	11.1%

layer with a 0.001 learning rate to benefit model convergence. Once the training is finished, the optimal weights of the encoder module are saved. Then the feature representations are generated by feeding original inputs into the optimal encoder module. The data visualization for this learning stage is illustrated in Figure 5.5. In each figure, the horizontal axis represents frequency, while the vertical axis represents time. Specifically, Figure 5.5a shows the comparison between the masked log-mel spectrogram inputs and the reconstructed outputs. This reflects the performance of the autoencoder model for reconstruction. Based on the best performance, the latent feature representations generated from the encoder module are shown in Figure 5.5b. We could see what feature representations look like compared with the original log-mel spectrogram inputs.

Table 5.4 : The parameters of the proposed autoencoder model

Layer Type	**Parameters**	**Output Shape**
Input	-	(216, 128, 1)
2D CNN	kernel=3 × 3, stride=1, output depth=128	(216, 128, 128)
2D Maxpooling	pool_size=2 × 2, stride=2	(108, 64, 128)
2D CNN	kernel=3 × 3, stride=1, output depth=64	(108, 64, 64)
2D Maxpooling	pool_size=2 × 2, stride=2	(54, 32, 64)
2D CNN	kernel=3 × 3, stride=1, output depth=32	(54, 32, 32)
2D Maxpooling	pool_size=2 × 2, stride=2	(27, 16, 32)
2D CNN	kernel=3 × 3, stride=1, output depth=32	(27, 16, 32)
2D Upsampling	size=2 × 2	(54, 32, 32)
2D CNN	kernel=3 × 3, stride=1, output depth=64	(54, 32, 64)
2D Upsampling	size=2 × 2	(108, 64, 64)
2D CNN	kernel=3 × 3, stride=1, output depth=128	(108, 64, 128)
2D Upsampling	size=2 × 2	(216, 128, 128)
2D CNN	kernel=3 × 3, stride=1, output depth=1	(216, 128, 1)

In the supervised learning stage, the temporal segment-level representations are assembled sequentially through the saved encoder module and then put into the BiLSTM model. The output units of the LSTM layers are set as 512 for forward and backward directions separately. After that, the dropout rate of 0.5 is applied. The final binary classification is obtained through the dense layer with the softmax activation. In this part, LSTM and GRU (Gated Recurrent Unit) models are also under consideration due to fewer parameters and training costs. However, the BiLSTM model could capture sequential information in both directions (see Section 4.2.3 for more details) and performs better in the experiment. On the other hand, the training cost for the BiLSTM model is checked: the training time for each epoch is generally 5s–21s and the number of epochs to converge is an average of 25. Based on this, the time cost is completely affordable. Therefore, we give priority to

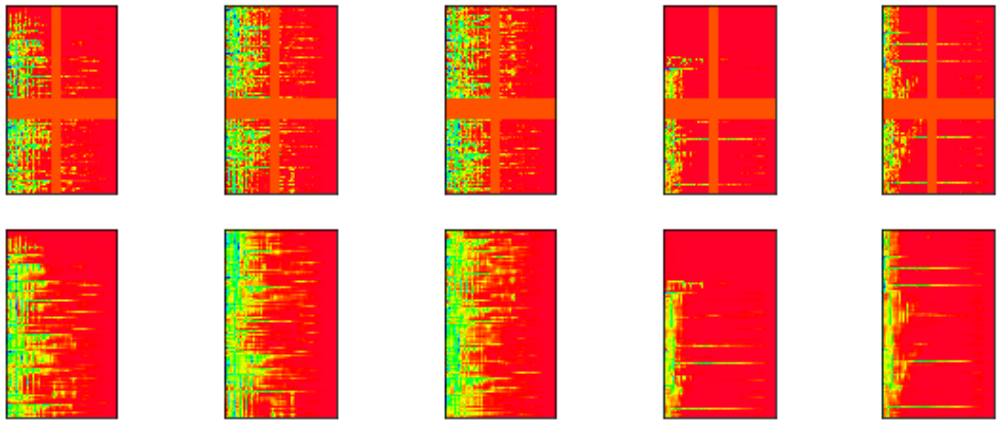
Table 5.5 : The hyper-parameters for model training

Hyper-parameter	**Unsupervised Learning**	**Supervised Learning**
Optimizer	Adam	Adam
Optimizer's Learning Rate	1e-3	1e-5
Batch Size	64	10
Loss	Huber	Categorical Cross Entropy

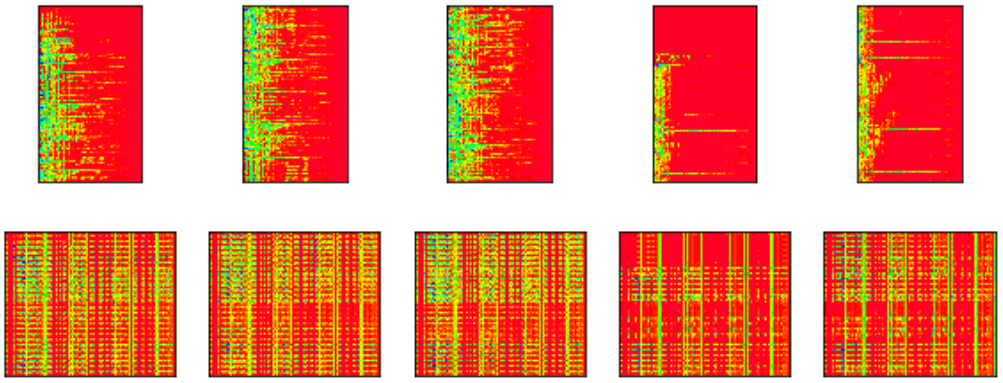
performance and choose the BiLSTM model.

The whole model is evaluated by running 10-fold cross-validation and obtaining the average performance based on classification accuracy and F1 score. Accordingly, the training/test sets are split with a ratio of 9:1. Specifically, the partitioning strategy is stratified random sampling based on valence labels to generate ten folds. For each pair of sets, 1 round of unsupervised learning (considering computing cost) and five rounds of supervised learning are conducted in the training set, and Valence/Arousal are predicted respectively in the test set to check the statistical results. For both unsupervised learning and supervised learning, the Adam optimizer (Kingma and Ba, 2015) is used, and the early stopping strategy is configured with the patience of 10-epoch for the validation dataset to avoid overfitting during training. The first-stage model monitors reconstruction Huber loss while the second-stage model monitors classification accuracy. The details of some hyper-parameters are summarized in Table 5.5. Moreover, the general time cost of two-stage model training on two datasets is reported (see Table 5.6). All experiments are implemented via Nvidia GeForce GTX 1080 GPU. The unsupervised learning usually takes 100–200 epochs per fold to converge. Supervised learning usually takes 20-30 epochs per fold to converge.

A baseline model is built to validate the advantage of the proposed autoencoder



(a)



(b)

Figure 5.5 : Data visualization in the unsupervised learning stage. We take five training samples and visualize the data before and after transformation. (a) shows the data change from the masked log-mel spectrogram inputs to the reconstructed outputs; (b) shows the original log-mel spectrogram inputs and their feature representations generated from the encoder module.

Table 5.6 : The general time cost of the proposed model during training

Dataset	**Segment Duration**	**Unsupervised Learning (CNN-based autoencoder)**	**Supervised Learning (BiLSTM)**
PMEmo	1s	75s/epoch, 3h/fold	21s/epoch, 525s/fold
	3s	24s/epoch, 1h/fold	13s/epoch, 325s/fold
	5s	15s/epoch, 0.6h/fold	7s/epoch, 175s/fold
	10s	8s/epoch, 0.3h/fold	5s/epoch, 138s/fold
AllMusic	3s	28s/epoch, 1.1h/fold	18s/epoch, 450s/fold
	5s	18s/epoch, 0.7h/fold	13s/epoch, 325s/fold
	10s	9s/epoch, 0.4h/fold	9s/epoch, 225s/fold

Note: In this table, 's' denotes second and 'h' denotes hour.

model, which combines CNN and BiLSTM directly. The CNN module reuses the structure of the feature encoder in unsupervised learning, followed by BiLSTM for emotion classification. These two parts are trained together.

5.4 Results

This section reports the experiment results based on selected segment duration, and the model performance is compared with previous work.

5.4.1 Performance of Different Segment Duration

The segments of different duration have been applied in the experiments. In multiple runs for each segment duration, the 10-fold scores are averaged. The results are shown in Table 5.7, and show that the performance for arousal recognition is always better than valence in all of the segment lengths investigated. The results also indicate that shorter segment length shows better performance on the valence dimension while longer segment duration benefits arousal performance. For example, in `PMEmo` dataset, the 1-second segment shows the best valence results with

79.01% accuracy and 83.2% F1-score while 5s/10s' segments show better accuracy (83.62%/83.51%) and F1-score (86.52%/86.62%) on arousal dimension. `AllMusic` dataset shows similar trends. For such results, the possible reasons are explained in the discussion section.

Table 5.7 : The performance comparison based on different segment duration

Dataset	**Segment Duration**	**Valence**		**Arousal**	
		Accuracy	***F1-score***	***Accuracy***	***F1-score***
PMEmo	1s	**79.01%**	**83.2%**	83.19%	86.1%
	3s	78.75%	82.95%	82.67%	85.59%
	5s	78.23%	82.64%	**83.62%**	**86.52%**
	10s	77.58%	82.18%	**83.51%**	**86.62%**
AllMusic	3s	**67.11%**	**67.11%**	85.67%	85.67%
	5s	66.89%	66.89%	**86.56%**	**86.56%**
	10s	66.45%	66.45%	86.11%	86.11%

5.4.2 Performance Comparison with Different Models and Sources

Table 5.8 shows a performance comparison with cutting-edge benchmarks based on different models and sources. This comparison shows that the proposed model can outperform any models using a single data source, either music or electrodermal activity signals. Compared to the model (Yin et al., 2019) that uses music sources only, the accuracy for valence prediction in the proposed model increases by more than 12% and the corresponding F1-score increases by more than 10%. Similarly, there are increases of almost 17% and 13% on arousal recognition in terms of accuracy and F1-score, respectively. The proposed model even competes with the latest multimodal framework (Yin et al., 2020) that utilizes EDA signals and music together with attention neural networks. Furthermore, the proposed model is compared with the baseline model, which uses segment-level inputs but lacks the autoencoder architecture. The results show that the two-stage model is superior to

the baseline model in both emotion dimensions.

Table 5.8 : The performance comparison with different models and different sources based on PMEmo dataset

Models	**Core Methods**	**Input Source**	**Audio Segmentation**	**Valence**		**Arousal**	
				Accuracy	***F1-score***	***Accuracy***	***F1-score***
RTCAN-1D (Yin et al., 2020)	attention module + ResNet + openSMILE	EDA + Music	No	77.30%	80.94%	82.51%	85.62%
RTCAG (Yin et al., 2020)	attention module + ResNet	EDA	-	63.61%	62.47%	64.05%	64.82%
SVM (Yin et al., 2019)	SVM	Music	No	70.43%	75.32%	71.49%	76.36%
SVM (Sharma et al., 2020)	SVM	Music + Lyrics	No	61.98%	-	68.75%	-
The baseline	CNN + BiLSTM	Music	Yes	77.44%	81.91%	82.79%	85.17%
Proposed model	CNN-based autoencoder + BiLSTM	Music	Yes	**79.01%**	**83.2%**	**83.62%**	**86.52%**

5.5 Discussion

5.5.1 Segment Duration Analysis

The performance in Table 5.7 indicates that a longer segment length contains more acoustic cues for arousal recognition, while a shorter one has less distracted information for valence prediction. Compared with segments of long duration, shorter segments are more likely to avoid changes in musical characteristics and reflect consistent perceptual properties of music like harmony and pitches that benefit valence recognition (Gabrielsson and Lindström, 2001). In contrast, the relatively long duration may capture more time-domain regularities like beat and tempo that benefit arousal recognition (Grekow, 2017). Further, the paired t-test is conducted to examine the performance of different segment duration. The results demonstrate that there is no statistical significance concerning which segment duration is best. The possible reason is that a log-mel spectrogram with the same input shape is used for different segment durations, which limits the selection of the FFT window and the hop length, thereby impacting the musical pattern extraction from audio data. Another reason is to what extent the segment duration could match the emotional boundary. Each fold contains songs with various emotional segmentation. The better performance depends on whether the fixed segmentation could cover emotional segmentation well for most songs (Aljanaki et al., 2015a). Generally, the 5-second segment is a relatively better choice for the proposed model as this duration is a reasonable trade-off between performance and computing cost.

5.5.2 Performance Analysis between Two Datasets

From Table 5.7, it also can be seen that the performance of valence recognition on `PMEmo` dataset is much better than that on `AllMusic` dataset. One reason is the dataset's peculiarity. All music excerpts in `PMEmo` dataset are the chorus portion of popular songs. It usually consists of an acoustic pattern that repeats. Due to this, emotional segmentation is more likely to coincide with those repetitive segments, which benefits emotion recognition (Aljanaki et al., 2015a). Another possible reason is the different ways to obtain annotation. The `AllMusic` dataset combined

social tags with the Warriner's VA ratings. Social tags have some problems, such as polysemy, misspellings and popularity bias (Lamere, 2008). The Warriner's VA ratings are based on word stimuli rather than music perception. Although the annotation was validated in a manual blind way, the deviation exists for some emotion-related terms. For example, 'black' and 'dark' are supposed to be linked with low valence, but their ratings are opposite in the Warriner's measurement. Therefore, such annotation maybe impact model prediction patterns.

5.5.3 Performance Analysis Compared with Other Models

In this part, the segment-based framework and model structures are discussed. Compared with the models in Table 5.8, the proposed model using segment-level learning shows better performance than other models that used the whole music excerpts directly. The long duration may contain acoustic cue variations and emotional state changes (Xiao et al., 2008), which may make learning models confused and have difficulties extracting unified musical features targeted to one kind of emotion (Aljanaki et al., 2015a). Segment-based learning relieves this problem as the relatively shorter duration usually reflects consistent music feature patterns that facilitate emotion recognition and improve the effectiveness of learning (Wu et al., 2014). On the other hand, two models with audio segmentation are compared. Under the same experimental circumstance, the two-stage model with the autoencoder structure outperforms the baseline model. It is demonstrated that the autoencoder can contribute to increasing final performance. The advantage is that the autoencoder separates two-stage training with their optimum parameters. In the meantime, no labels are required as an unsupervised learning method. Further, segment-level unsupervised learning brings about more flexibility in the model structure design. The framework is divided into two parts. One part concentrates on feature representation, while the other part focuses on target prediction. It is possible that one part could be replaced without changing the other part as long as the data interfaces could match well with each other meaningfully. For example, another effective deep neural network predicts the final emotion instead of the LSTM model. Other advanced learning models could be considered in future research.

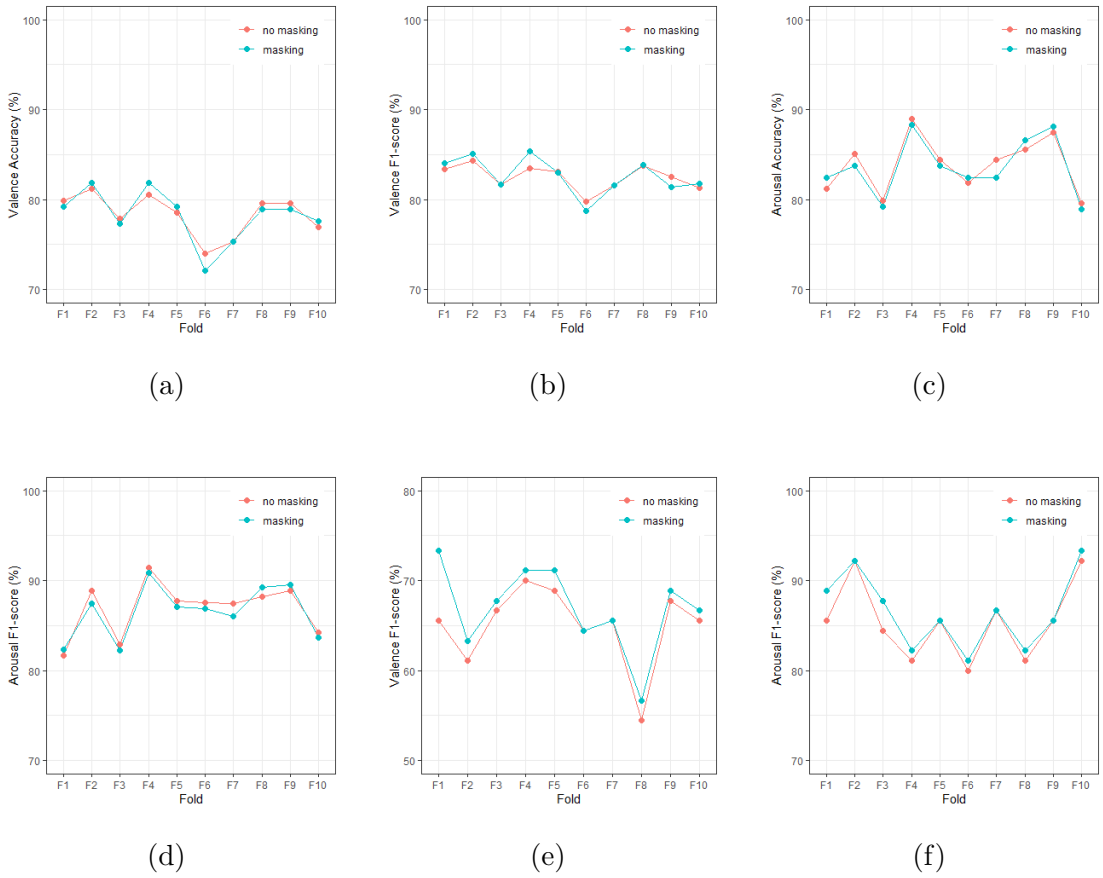


Figure 5.6 : Masking impact on the performance. For `PMEmo` dataset: (a) shows the accuracy for valence; (b) shows F1-score for valence; (c) shows the accuracy for arousal; (d) shows F1-score for arousal. For `AllMusic` dataset: (e) shows F1-score for valence; (f) shows F1-score for arousal; the accuracy comparison is same as F1-score.

Another factor to be considered is the cost. The state-of-the-artwork adopted attention mechanisms (Yin et al., 2020). This is powerful for learning music representations, but it introduces more training parameters and increases the complexity of computing, which requires more computing resources and aggravates the burden of the operating environment, even extra time cost (He and Sun, 2015). My design replaces the attention architecture with stacked convolutional neural networks, which reduces the time cost (refer to Table 5.6) but achieves the equivalent results. We argue that the model structure in this research is more cost-effective.

5.5.4 Ablation Test for Masking Data

The ablation test is carried out to examine the effectiveness of the masking method. The 10 folds of accuracy and F1-score for valence/arousal recognition are visualized in Fig. 5.6. In each subfigure, both lines represent the model's performance without masking and the model with masking. For `PMEmo` dataset, both lines cross several times. The characteristics of the dataset can explain this result. The chorus part of a popular song contains the repetition of musical content that shows more explicit and more intense emotional expression (Yeh et al., 2014). Such data morphology decreases the data variation and the outliers to lessen the effect of masking methodology. For `AllMusic` dataset, it contains different genres of songs and balanced training samples. The effectiveness of masking is statistically significant. Overall, it is believed that masking could benefit the model's robustness. In future work, we may investigate the effect of different proportions of masking spans on performance.

5.6 Summary

In this chapter, a segment-level two-stage learning framework is proposed. This naturally combines unsupervised learning as a feature extractor with supervised learning as a music emotion classifier. First, a CNN-based autoencoder calculates feature representations for contiguous segments that comprise each music excerpt. And then, the time-series segments are fed into the BiLSTM model to predict emotion for this music excerpt. In this way, segment-level features are extracted without being limited to song-level annotation. Additionally, the time/frequency masking approach is applied to the segment inputs to enhance model robustness. The experimental results show that the proposed model performs better than those using a single feature source, even competing with the cutting-edge multi-modal framework. Compared with the whole music excerpts as model inputs, segments with relatively short duration increase the data scale and contain less change of acoustic cues. Due to this, the learning models could detect the correlation between musical features and emotion more effectively. Apart from that, this two-stage training framework

is more flexible and makes changing the combinations of neural networks possible. That means much potential for performance improvement. This chapter has resulted in a publication (He and Ferguson, 2022), thus validated by peer-reviewers already.

Chapter 6

Music Social Tags Representation in Dimensional Emotion Space

6.1 Introduction

In music emotion recognition research, much research has sought to recognize and retrieve music based on emotion labels. These labels are usually obtained from either subjective tests or crowd-sourcing resources. With the widespread use of social media, social tags are ready-made data and are a good option to extract emotion annotation for music training models. Researchers usually used emotion-based social tags either by grouping tags into emotion categories or clusters directly, or by mapping tags to dimensional quadrants simply. Few research has undertaken text-based analysis covering more kinds of social tags (not only emotion tags) to explore the relationship between tags and represent them in a dimensional emotion space, especially through neural word embedding techniques and the large-scale dataset.

To narrow that research gap, this study proposes a solution for tags analysis and representation based on neural word embedding methods. The results show that these methods outperform traditional semantic analysis methods. This solution can model joint representations of tags rather than be limited to a single type of tag corpus (such as emotion or genre only) and quantify social tags in dimensional emotion space. This might be utilized as emotion annotation for music.

6.2 Methodology

This section describes how to process tags information and represent them in dimensional emotion space. Figure 6.1 shows the overview of the solution. The social tags dataset is preprocessed to filter out some redundant information and prelimi-

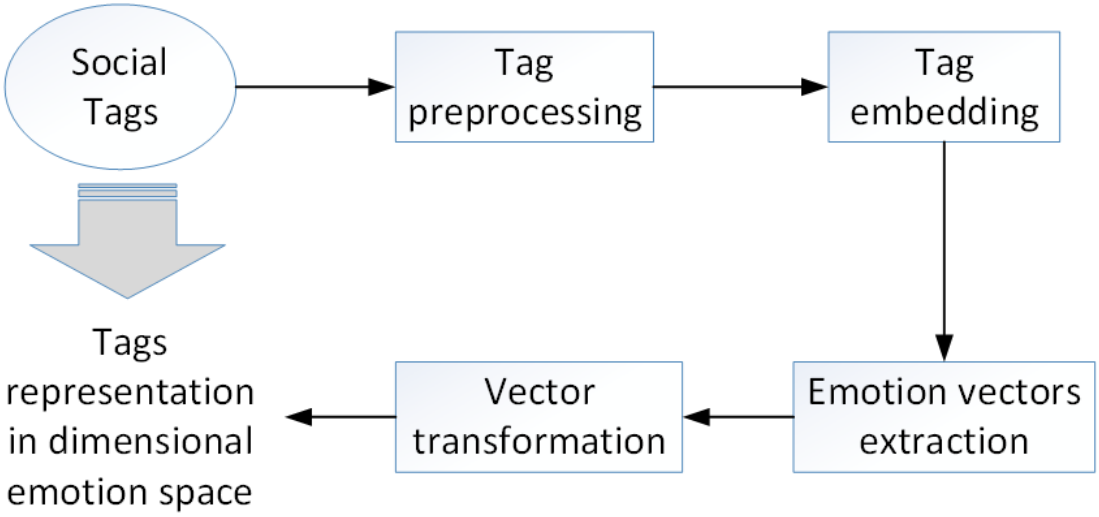


Figure 6.1 : The overview of tags analysis solution

narily converted into a text corpus or the factorized matrix. Then tag embedding models are applied to the structured data to explore the latent feature vectors for tags. Once the vector-based tags are ready, typical emotion tags are extracted to match the reference emotion plane later. Then these vectors with high dimensions are transformed into low-dimension vectors that are represented in a dimensional emotion space. In this workflow, tag preprocessing and tag embedding play significant roles in reducing the sparsity of social tags. The selection of emotion tags and the emotion measurement criteria determine the quality of tag representation to some extent.

6.2.1 Tag Preprocessing

A large-scale set of social tags is collected from `Last.fm`* which is combined with the Million Song Dataset (MSD) (Bertin-Mahieux et al., 2011) and has been used in many music classification research projects (Hu and Downie, 2007; Laurier et al., 2009; Song et al., 2016). Based on previous research (Lamere, 2008), it is necessary to preprocess tags to reduce the impact of noisy information and irrelevant information in this tag dataset.

*http://www.last.fm

The first step is to construct a text corpus to facilitate tag preprocessing. To describe the solution comprehensively, each track in the songs dataset is viewed as a document and tags for one track are viewed as text in one document. Besides that, each tag is defined as a "term", not a "word" since not all tags are single words. Like term frequency in documents, `Last.fm` dataset contains tag popularity for tracks. Some researchers used these normalised counts to calculate Term Frequency-Inverse Document Frequency (TF-IDF) (Saari and Eerola, 2014; Levy and Sandler, 2008). This solution combines these normalised counts and corresponding tags to build up tag content for each track to construct a text corpus for all tracks. In detail, each file in `Last.fm` dataset represents one song identified by its "track_id". The file content is JSON-encoded. For example, the "tags" part shows each tag with its popularity for one song like this:

"tags": [["happy", "5"], ["pop", "2"], ["powerful", "1"]]

Then the tag document for this song is organized like this:

"happy happy happy happy happy pop pop powerful"

Once the text corpus for all songs is ready, tags are categorized to determine what strategies should be applied to this corpus to process different types of tags. Based on previous research work (Hu et al., 2009; Saari and Eerola, 2014; Çano and Morisio, 2017b), music social tag categories are summarized with examples as below:

- **meaningless terms:**

  stop words: a, the, this, no, not
  junk tags or misspellings: zzzzzzz, Grrl

- **non-emotion terms:**

  opinion words: good, bad, poor
  genre, instrument, epoch, locale: jazz, guitar, 60s, usa
  ambiguous tags: love
  emotion-irrelevance tags : song, beat

- **emotion terms:**

  lemmatization: depression, depressive, depressed

  synonym: melancholy and sadness

Meaningless Terms

For the stop words, most of them are meaningless for semantic analysis but take a high proportion in the corpus. Due to this, stopwords are removed by referring to the `snowball`[†] list of stopwords. At the same time, negative words in this list remain thereby keeping the original meaning for some terms such as 'not happy'. The junk tags and misspellings should be removed to improve the validity of the tags. Considering the variety of tag content, it is impossible to find out all tags mentioned above and filter them manually. Supposing that these terms are either very common or low-frequency, a series of statistical thresholds are set to filter them:

- **term_count_min:** minimum number of occurrences over all documents
- **doc_proportion_max:** maximum proportion of documents that should contain the term
- **doc_proportion_min:** minimum proportion of documents that should contain the term

In this way, most meaningless, noisy, high-frequency terms could be excluded. To some extent, the thresholds determine the quality of term analysis to balance between removing irrelevant information and avoiding information loss.

Non-emotion Terms

Previous research work (Lamere, 2008) explored `Last.fm` tags dataset and found that tags mainly include genre, emotion, instruments, locales, opinions and so on. Among them, genre accounts for a high proportion (68%) followed by locale (12%) while mood only accounts for 5% followed by opinion (4%) and instruments (4%).

[†]http://snowball.tartarus.org/algorithms/english/stop.txt

Focusing on emotion tags analysis, most research usually removes all non-emotional terms and only keeps tracks labelled by emotion tags. Such approaches result in a great deal of information loss and lack of generalization since a large number of tracks without emotion tags are excluded. In my research, more tracks are involved in a subsequent tag embedding model so that more tags relationship could be explored in vector space. Using this method, even if a track is not labelled by emotion tags, it could be linked to emotion terms through term similarity and analogy. For ambiguous tags and other emotion-irrelevance tags, most of them are excluded through statistical filtering mentioned in Section 6.2.1

Emotion Terms

Considering the inflection of words and synonyms, some researchers (Hu et al., 2009) tried to build up synsets for clustering emotion terms while others (Çano and Morisio, 2017b) extended term inflected forms derived from a lemmatization process to construct emotion corpus. In my research, no change is made for all emotion terms so that we could explore whether these terms have distinct dimensional values from each other.

After the preprocessing mentioned above, the final tag corpus is established and then a corpus of textual data representing all tracks is vectorized for further use.

6.2.2 Tag Embedding

Corresponding to different word embedding models, different types of input are required and constructed from the vectorized text corpus mentioned above. The following parts introduce a series of model algorithms for tag semantic analysis.

Latent Semantic Analysis

The conventional Latent Semantic Analysis (LSA) technique requires a Document-term matrix (DTM) as input and tries to find a low-rank approximation to the term-document matrix. DTM describes the frequency of terms that occur in a collection of tag documents. To reduce the impact of high-frequency terms, the term-weighting scheme TF-IDF (Wu et al., 2008) is usually applied to adjust term

weights. The rank-lowering process is based on the theory of linear algebra called SVD (singular value decomposition), which conducts the matrix factorization. The new low-dimensional term-document matrix is viewed as a set of term vectors. That could be used to explore the relationship between terms (tags) or documents (songs) for many purposes. In my research, LSA is used as a baseline.

CBOW and Skip-gram

Neural word-embedding models CBOW (Continuous Bag-Of-Words) and skip-gram take the vectorized text corpus as input directly rather than using global matrix factorization. These two models can learn the local context for each term automatically. The CBOW model predicts the target term according to its context. On the contrary, the skip-gram model learns to predict the surrounding terms from a given target term. Both models are neural networks with one hidden layer embedded. The optimal weights of the hidden layer during training are the term vectors we want to use.

As shown in Fig. 6.2, the CBOW model takes N co-occurred terms as inputs and each term is an encoded vector of size V based on tags corpus. The hidden layer contains K-dimension neurons and the output is the target term of size V through softmax calculations. While the operation of the Skip-gram model is reversed. The input layer is a target term of size V and the output is the probability distributions of N context terms based on this target term. Similarly, the hidden layer is made of K-dimension neurons. For both models, the goal of model training is not for prediction, but for obtaining the weight matrix $W_{V\times K}$. This weight matrix is viewed as tag embeddings that represent V terms with the K-dimension feature vector for each.

GloVe

GloVe is an unsupervised learning model. The model is trained based on the term-co-occurrence matrix (TCM). TCM is the statistics of terms in the vectorized corpus in a form of matrix X. Each element X_{ij} in such matrix represents how often term i appears in the context of term j. The algorithm utilizes a new weighted least

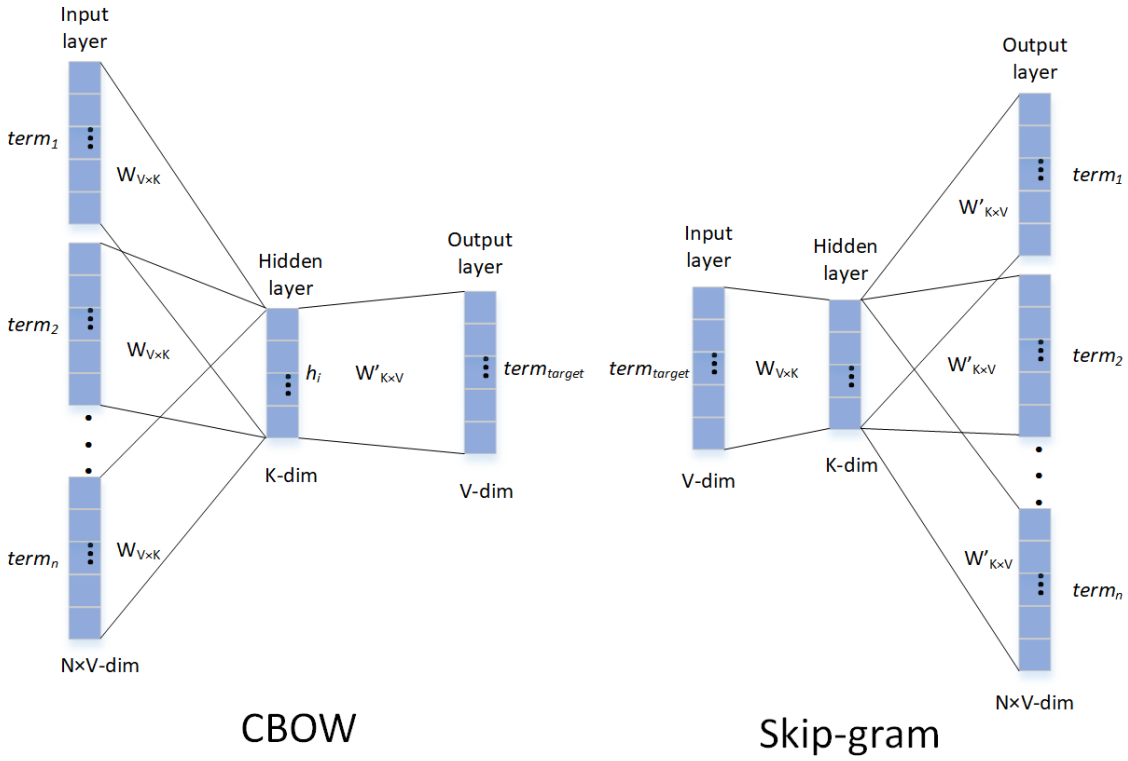


Figure 6.2 : Neural network structure for tag embedding

squares regression model (Pennington et al., 2014). It defines a cost function like this:

$$J = \sum_{i=1}^{V} \sum_{j=1}^{V} f(X_{ij})(w_i^T w_j + b_i + b_j - log(X_{ij}))^2 \tag{6.1}$$

Here w_i means the vector for the main term i and w_j means the vector for the context term j. b_i and b_j are scalar biases for the main and context terms. f is a weighting function that avoids frequent co-occurrences of being overweight, see the definition as:

$$f(X_{ij}) = \begin{cases} (\mathrm{X}_{ij}/X_{max})^{\alpha} & \text{if } \mathrm{X}_{ij} < X_{max} \\ 1 & \text{otherwise} \end{cases} \tag{6.2}$$

Here X_{max} defines the threshold of term co-occurrences value. Only X_{ij} less than X_{max} take effect to regression model through f. α is a factor in the weighting function, set to 0.75 by default.

The GloVe model decomposes TCM into two low-rank matrices that are two sets of term vectors called main term vectors and context term vectors. In practice, the

final term vectors are the average or a sum of these two vectors.

Through tag embedding, all of these models output a vector-based matrix, where

- each row presents each term in the corpus of tags
- all elements in each row is a feature vector for that term
- the vector size means dimensions of tag embedding

6.2.3 Emotion Vectors Extraction

Once term vectors are ready, vectors for specified terms could be extracted to analyse the tags' relationship and check the tag-embedding model performance. A set of terms can belong to one specified type such as emotion, genre or theme, which reflects the generalization of our solution. In this thesis, social tags analysis is based on emotion representation, hence the emotion vocabulary is defined for vector extraction and subsequent mapping.

6.2.4 Vector-based Data Transformation

In this step, the non-metric multidimensional scaling (nMDS) method and Procrustes analysis are applied to emotion term vectors for vector-based dimensionality reduction and transformation. Dimensional emotion models proposed in the previous research are usually two or three dimensions with the reason that more than 3 dimensions could not reflect emotion variation intuitively and one dimension could not distinguish emotion sufficiently. In this research, nMDS (Kruskal, 1964) is utilized to generate 2D and 3D models separately for performance comparison. Then Procrustes transformation (Gower, 2015) make emotion tags approximate to a classic Valence-Arousal (VA) model.

To assess the quality of nMDS, nMDS is compared with other typical dimensionality reduction solutions including Principal component analysis (PCA), Locally Linear Embedding (LLE), kernel PCA (kPCA) and AutoEncoder, through R_{NX} measurement defined in (Kraemer et al., 2018). Taking a set of our vector-based data as input, the R_{NX} values with log-scaled rank K are shown in Fig. 6.3. The

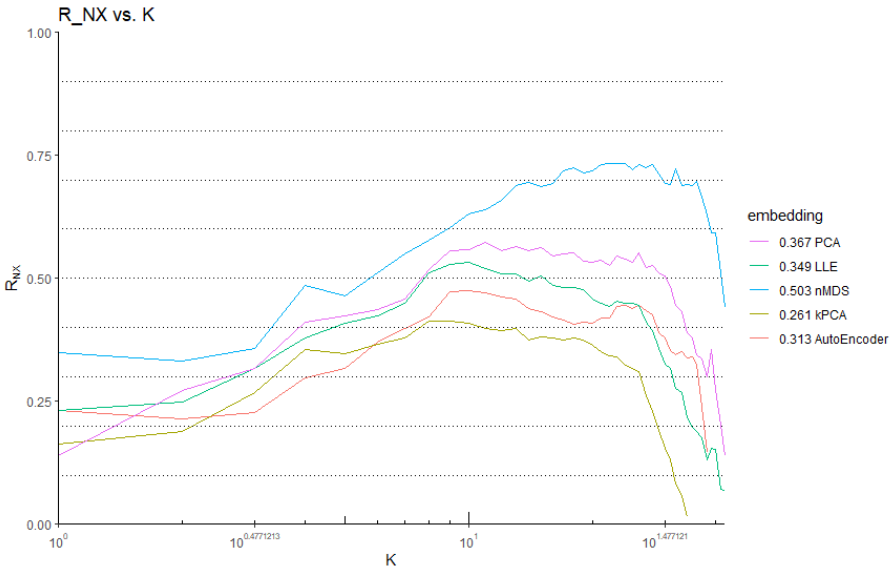


Figure 6.3 : Performance comparison of different dimensionality reduction methods, where a value of 0 corresponds to a random embedding and a value of 1 to a perfect embedding into the k neighbourhood. The legend contains AUC_{lnk} measurement defined in (Kraemer et al., 2018).

results show that nMDS is the best way to represent the pairwise distance and dissimilarity for terms in low dimensions meanwhile keeping the pairwise relationship changing as few as possible.

In the procedure of Procrustes analysis, the 2D and 3D term vectors are further transformed through translation, rotation and scaling to find an optimal approximation to the VA reference. Given the VA reference as target X, our vectors as Y are transformed to conform to X. Y_t is the final 2D emotion tags representation. Equation (6.3) shows how PA works.

$$Y_t = f(Y) = b * Y * T + c \tag{6.3}$$

where b is scale component, T is orthogonal rotation and reflection component, and c is translation component.

To obtain the better goodness-of-fit, b, T and c are adjusted to minimize the

root mean squared error (RMSE) defined below:

$$f(Y) \rightarrow \sqrt{\frac{\sum_{min}(X - Y_t)^2}{N}} \tag{6.4}$$

where N means the number of terms in the vector-based data.

If Y is a 3D vector, then 2D VA reference X is filled with one column of zero to match dimensions. Alternatively, the dominance ratings (Warriner et al., 2013) could be used as the third dimension but this situation is not covered here.

6.3 Experiments

Corresponding to the workflow mentioned in Section 6.2, The experiment details are described step by step. Tags processing and transformation are implemented by R language. The code source could refer to the GitHub[‡].

6.3.1 Dataset Preprocessing

The social tags dataset is collected from `Last.fm` associated with 504,555 tracks in Million Song Dataset (MSD). After removing stop words, a vocabulary is created containing a total of 463,487 unique tag terms. Then very common terms and low-frequency terms are filtered out based on statistical thresholds. Fig. 6.4 shows how many terms are left with different combinations of thresholds. From the view of the minimum proportion of documents containing terms, it shows differences starting from 0.0002. Then combined with the view of the minimum number of occurrences over all documents, it changes obviously since 1000. For the maximum proportion of documents containing terms, not too much gap exists between 0.1 and 0.8. All of these parameters determine the number of remaining terms. If the scale is too big, it might lead to a relatively large deviation in dimensional reduction. By experimenting with parameters and analyzing the results, Table 6.1 is the final set to balance information noise with information loss. This results in a total of 7,685 terms and a corpus of 470,280 tracks. Finally, this corpus is vectorized for the

[‡]https://github.com/Sandy-HE/Tag_analysis

subsequent process. All preprocessing is implemented through the R development environment.

6.3.2 Tag Embedding Model Setup

In the process of tag embedding, embedding dimensions K need to be considered. Given that the final emotion model is 2 or 3 dimensions, the higher K values would increase the dimension gap and may give rise to underfitting by leaving out important dimensions of the dissimilarity data when implementing dimensionality reduction. Due to this, K is selected from the set of {4, 8, 16, 32, 64, 128}.

In this experiment, the natural language processing package `text2vec`[§] in R software environment is used to generate pruned corpus, DTM with TF-IDF, TCM and run LSA and GloVe models. CBOW and Skip-gram models are implemented by `word2vec` function in python library `gensim`[¶]. Table 6.2 lists the detail of input and some key hyper-parameters used in embedding models. For other parameters, the default values are used as defined in the functions.

Table 6.1 : Thresholds of term filtering

Parameter	**Value**
term_count_min	1,000
doc_proportion_max	0.8
doc_proportion_min	0.0002

6.3.3 Emotion Terms Selection

The selecting of emotion vocabulary is based on dimensional emotion models (Russell, 1980; Scherer, 2005; Saari and Eerola, 2014), emotion clustering (Hu et al., 2009; Laurier et al., 2009), and MIREX Mood Categories (Hu et al., 2008). In

[§]https://cran.r-project.org/web/packages/text2vec/

[¶]https://pypi.org/project/gensim/

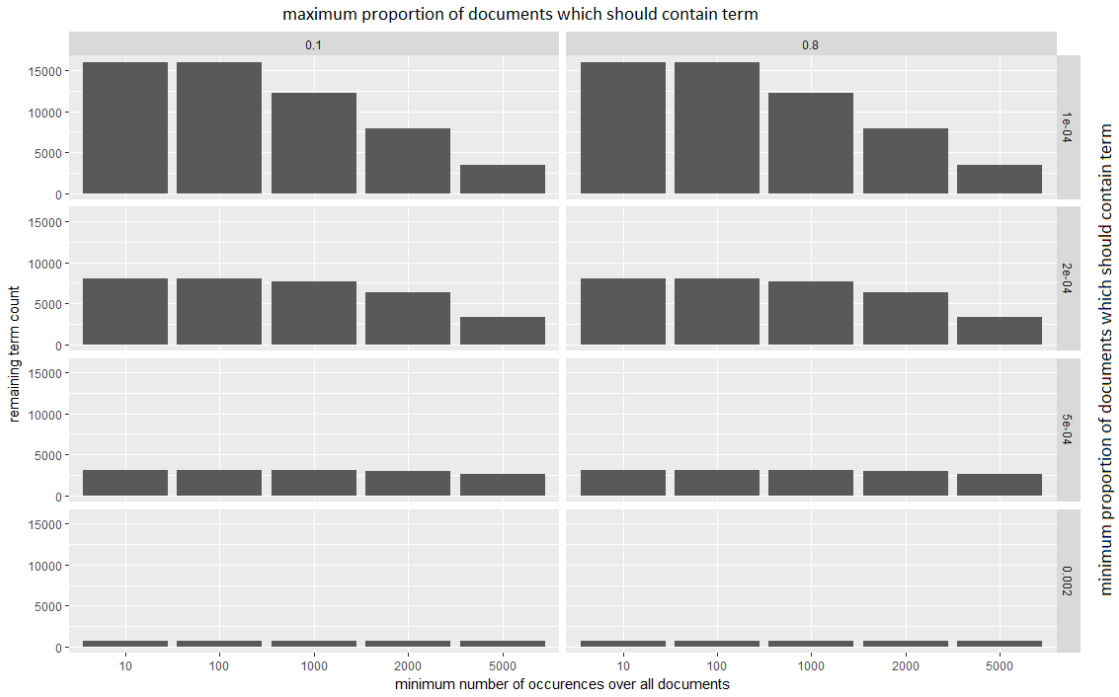


Figure 6.4 : Terms count statistics after filtering

my experiment, the initial term list was established according to the mood clusters summarised by Laurier et al. (2009). Then some terms were crossed out because they might not apply exactly as before, such as "gay". To balance terms in each quadrant, other terms are added regarding research in recent decades mentioned above. At the same time, it should be checked that the selected terms are in the set of social tags and show appropriate positions after data transformation based on experimentation. Finally, 44 common emotion terms are selected as the benchmark, corresponding to VA emotion quadrants considering term balance in each quadrant (see Table 6.3).

6.3.4 Data Transformation

In this step, the nMDS method is applied to reduce K-dimension vectors to 2D and 3D vectors, respectively, followed by Procrustes transformation. In Procrustes analysis, the target VA references are chosen from the Warriner's list (Warriner et al., 2013), which provides continuous ratings of valence, arousal and dominance for 13,915 English words. Still, all data transformations are implemented under the

Table 6.2 : Tag-embedding models summary

Model	**Input**	**Hyper-parameters for embedding**
LSA	DTM with TF-IDF	vector size = K
CBOW	Corpus	vector size = K context window = 5 training epoch = 10
Skip-gram	Corpus	vector size = K context window = 5 training epoch = 10
GloVe	TCM (context window = 5)	vector size = K $X_{max} = 10$ training epoch = 25 learning rate = 0.15

R development environment.

6.4 Results and Discussion

This section compares the performance of neural tag embedding models and GloVe with the LSA baseline. Further, tag topology structure based on dimensional emotion space is visualized, and the influence factors for tag representation are discussed.

6.4.1 Tag Embedding Models Performance

The performance comparison between four tag embedding models is shown in Fig. 6.5 based on Procrustes RMSE as the evaluation metrics. It illustrates that better performance is located in the higher K range {32, 64, 128} for all models because that values narrow the gap between the sparse high-dimension corpus and

Table 6.3 : The emotion tags in dimensional quadrants

Q1	**Q2**	**Q3**	**Q4**
happy	angry	sad	relax
joyful	brutal	melancholy	calm
party	aggressive	sadness	peaceful
fun	scary	depressive	mellow
sexy	frustration	bittersweet	sweet
upbeat	bitter	gloomy	soothing
uplifting	sarcastic	sorrow	hopeful
exciting	cynical	desperate	dreamy
triumphant	black	dark	chill
intense	quirky	lonely	serious
romantic	heartbroken	sleepy	quiet

K-dimension embedding vectors. The best solution for each K-dimension embedding is one of the neural tag embedding models rather than LSA. This demonstrates that neural word embedding techniques outperform conventional text analysis methods. Further, GloVe and Skip-gram models' performance varies dramatically with K value changing while the performance of the CBOW model is relatively stable. It can be seen that GloVe and Skip-gram models are more sensitive to the embedding size, and hence selecting the appropriate size could achieve better performance. There is no certain regularity to impact performance in selecting 2D or 3D vector space. In my experiment, the best results are based on 64-dimension tag embedding, where the vectors from the GloVe model are reduced to 2D while the vectors from the Skip-gram model are reduced to 3D.

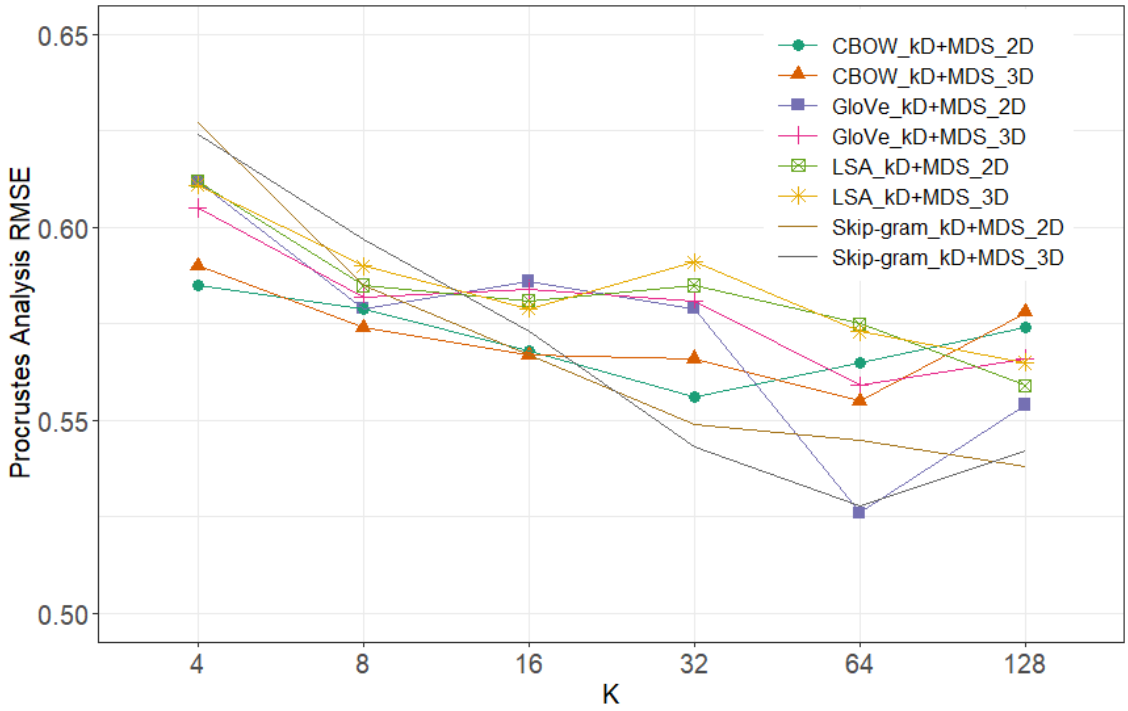


Figure 6.5 : Procrustes analysis performance comparison between different models

6.4.2 Tags Visualization

In this part, the transformed results from Procrustes analysis are used to visualize the final 2D emotion space based on the social tags relationship. Here two models with the best performance are selected: *GloVe_64D+MDS_2D* and *Skip-gram_64D+MDS_3D*. The results are shown in Fig. 6.6 and Fig. 6.7. In these two emotion models, it can be seen that tags could be grouped into four quadrants conforming to the typical VA emotion model. Tags representation based on the Skip-gram model shows the typical terms better than the one based on the GloVe model, such as happy, angry, sad, and relax. The deviation of 'sad' in Fig. 6.6 is caused by GloVe combining global matrix factorization. In such latent semantic analysis, many co-occurrence terms with 'sad' are non-emotion terms, but these terms co-occur with other high-frequency emotion terms, which are not located in Q3 but in Q4. While the Skip-gram model in Fig. 6.7 only utilizes local context information, reflecting the latent relationship with 'sad' better and reducing the impact of noise information. It illustrates that terms in the social tag corpus are strongly correlated

with the terms nearby with similar popularity, and cleaning irrelevant information is very important for GloVe models as it covers the global context. Skip-gram is the better choice without cleaning the corpus on a large scale.

Another reason for the position deviation is the calculation of the Procrustes analysis. It tries to minimize the sum of residuals for all tags between term vectors and VA references. To serve the whole performance, some terms sacrifice their correct positions to bridge the gap of inappropriate positions of terms influenced by irrelevant information, such as angry and aggressive in Fig. 6.6. Besides that, the Warriner's VA ratings bring about some deviations. The difference exists for some terms because those ratings are based on word stimuli rather than music. For example, 'heartbroken' in music emotion models are supposed to be located in low arousal, but its reference is labelled in high arousal. Similarly, 'black' and 'dark' are usually linked with heavy metal music and express low valence and middle or high arousal, but their ratings are opposite in VA reference. Therefore, a better VA reference could enhance the transformation performance dramatically. More details can be found in Appendix A.

6.4.3 Music Emotion Annotation based on Tags

In Music Emotion Recognition tasks, social tags are used for music annotation in several ways (Çano and Morisio, 2017b; Panda et al., 2018; Laurier et al., 2009). But most prior research applied these tags to solve music classification problems rather than regression problems because there is no good way to quantify tags and then quantify emotion for songs. In contrast, my solution provides a more flexible and effective way to represent tags as embedding vectors to map them into dimensional emotion space. On the one hand, the quantified emotion for music based on tags could be provided as mentioned in (Saari and Eerola, 2014). On the other hand, annotating songs without labelled emotion tags in a dimensional emotion space could be considered by exploring the tags relationship, such as tags semantic analysis and tags analogy. It is more reasonable to get emotional annotation for more songs.

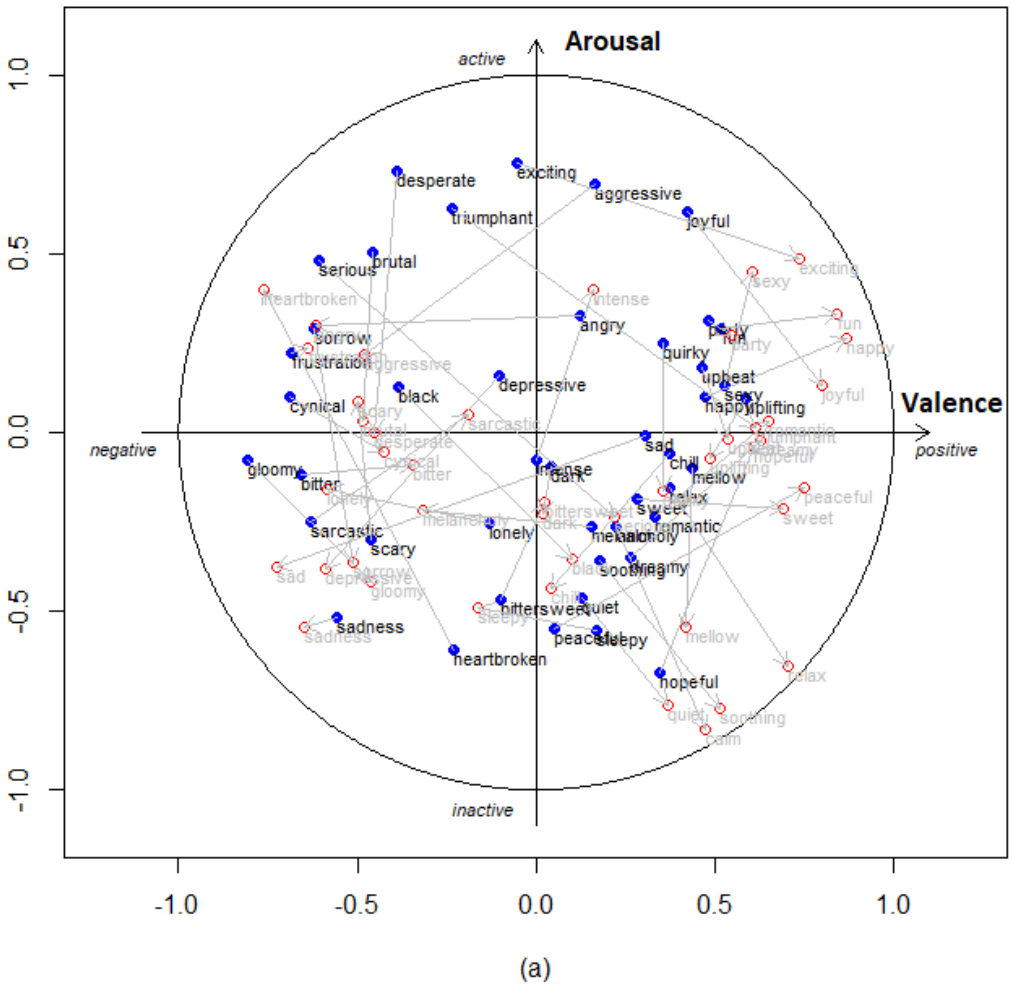


Figure 6.6 : Tag representation based on GloVe model. In detail, the data flow includes 64D tag embeddings from the Glove model, 2D tag vectors from nMDS, and a final approximation from Procrustes analysis. In this figure, blue dots represent social tag positions, and red circles represent Warriner's VA reference of tags.

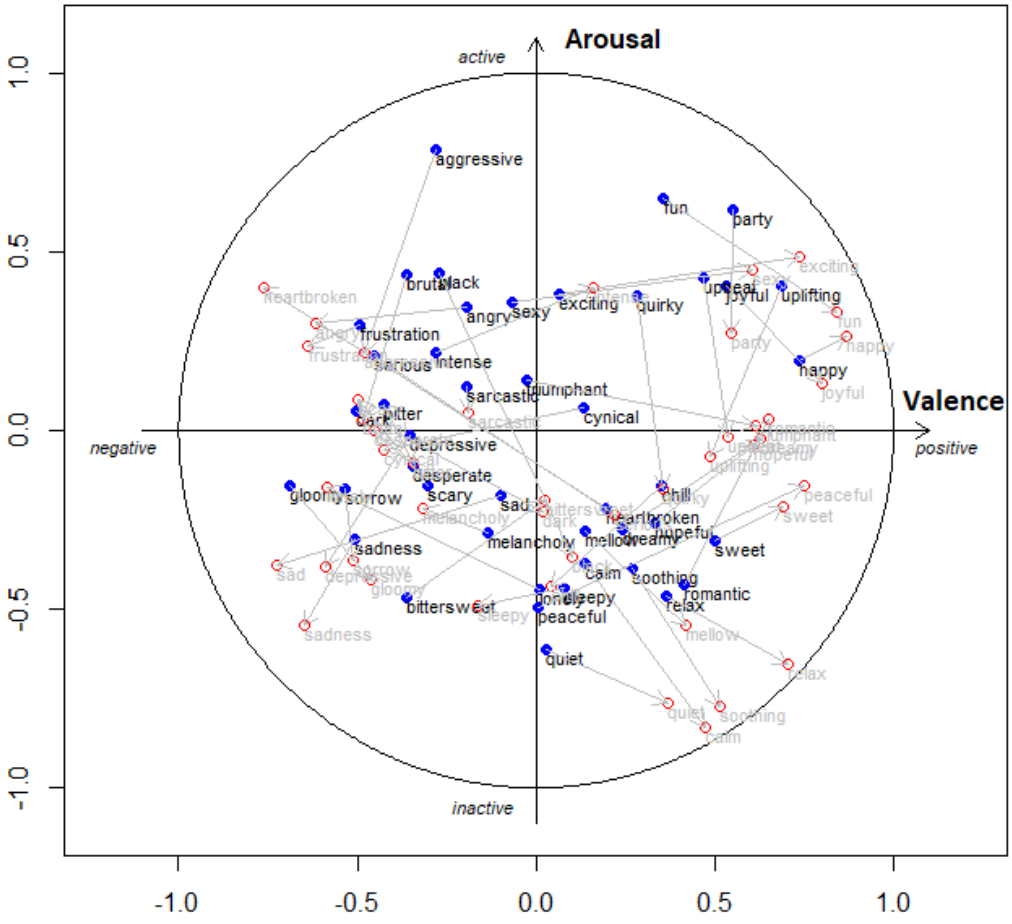


Figure 6.7 : Tag representation based on Skip-gram model. In detail, the data flow includes 64D tag embeddings from the Skip-gram model, 3D tag vectors from nMDS, and a final approximation from Procrustes analysis. In this figure, blue dots represent social tag positions, and red circles represent Warriner's VA reference of tags.

6.5 Summary

This research proposes an effective solution to analyse the music social tags relationship, where the neural word embedding models are applied to obtain vector-based terms for large-scale tags datasets. Apart from that, non-metric MDS and Procrustes transformation quantify terms in a 2D VA emotion model. The experimental results show that neural tag embedding models outperform the conventional LSA model. The solution transforms the sparse, discrete and messy tags information into dense, quantified and correlated data. In this way, it allows more than one type of term (e.g. genre, emotion) in the corpus for tag analysis and can reflect tags' relationship covering multiple vocabulary sets. Moreover, the semantic analysis could link songs without labelling emotion tags with emotion terms. This provides a good resource for emotion annotation in music emotion recognition work.

Considering how to adapt this approach to MER tasks, Saari and Eerola(2014) gave some extensions based on similar quantified tags. No ready dataset containing audio data matched with such large-scale social tags exists. Million Song Dataset provided some audio features metadata, but the expected data in my research is music audio. A considerable effort is needed to collect such raw data and extract appropriate music clips. Besides data collection, the critical research point is calculating music emotions based on their tags. Many factors should be taken into account, such as using a full or partial set of tags for one song, along with tag popularity or not, etc. Also, the design for tag denoising should be optimised to promote quantitative quality. The future study might cover these tasks. This chapter has resulted in a publication (He and Ferguson, 2020a) , thus validated by peer-reviewers already.

Chapter 7

Conclusion

7.1 Summary and Conclusions

This thesis focuses on music emotion-related research with neural network applied, mainly including music emotion recognition (MER) by using deep learning models and social tags representation in the dimensional emotion space through neural word embedding models. In previous research, some traditional study patterns and methodologies could be breakthroughs: using human-engineered audio features as learning model inputs; using the whole music excerpts as model inputs to match the given annotations without considering music segmentation or limited to segment-level annotation; using music social tags discretely without exploring tags relationship through text-based analysis or neural word embedding applied. Correspondingly, this thesis set out three objectives: *i*) design a deep neural network model that is able to use raw audio signal data as training inputs directly instead of using preprocessed audio features and achieve better performance; *ii*)propose a deep learning architecture that could accept segments partitioned from the given music clips as model inputs without collecting extra segment-level annotations and complete final emotion recognition. *iii*) provide a solution for music social tags analysis with neural word embedding models applied to represent tags in a dimensional emotion space.

To better understand these studies, this thesis introduces the background knowledge in Chapter 2 and related work in Chapter 3.

In MER tasks, the research achievements are validated in two MER scenarios: one scenario detects dynamic emotion variation covering the first objective; the other scenario is conducting static emotion classification covering the second research objective. For social tags analysis, four tag embedding models are implemented to

compare the performance of dimensional emotion representation, which commits to the third objective. This thesis introduces methodologies, experiment details, results, and discussion for each goal.

In Chapter 4, an end-to-end deep learning approach is proposed to address music emotion recognition as a regression problem, using the raw audio signal as input (to achieve Objective i). The multi-view convolutional neural networks as feature extractors are adopted to learn representations automatically. Then the extracted feature vectors are merged and fed into two layers of Bidirectional Long Short-Term Memory to capture temporal context sufficiently. In this way, my model can recognise dynamic music emotion without requiring too much workload on domain knowledge learning and audio feature processing. The experimental results show that my model outperforms the state-of-the-art baseline with a significant margin in terms of R^2 score (more than 10%) on the `emoMusic` Dataset.

In Chapter 5, a segment-based two-stage model is proposed to combine unsupervised and supervised learning. In the first stage, an unsupervised autoencoder with stacked deep neural networks generates feature representation (to achieve Objective ii). In the second stage, these features are fed into a supervised learning model to predict emotion. Based on the architecture mentioned above, segmentation is applied. Each music excerpt is split into contiguous segments. In the unsupervised learning stage, these segments are fed into the autoencoder to extract segment-level feature representations. In the supervised learning stage, these time-series music segments as the whole inputs are fed into a Bidirectional Long Short-Term Memory deep learning model to achieve the final music emotion classification (to achieve Objective iii). Compared with the whole music excerpts, segments as model inputs could be the proper granularity for model training and augment the scale of training samples to reduce the risk of overfitting during deep learning. Also, masking frequency and time are applied to segment-level inputs in the unsupervised learning part to enhance training performance. The proposed model is evaluated on two datasets: `PMEmo` and `AllMusic`. The results show that the proposed model outperforms state-of-the-art models, some of which even use multimodal architec-

tures. And the performance comparison also evidences the effectiveness of audio segmentation and the autoencoder with masking in an unsupervised way.

Chapter 6 proposes a tag analysis solution for dimensional emotion representation. This solution includes social tags preprocessing, tag embedding, dimensionality reduction for tag feature vectors and vector transformation to approximate VA emotion ratings. The crucial part of this solution is tag embedding with neural network algorithms applied. Four tag embedding methods are compared. The results show that neural tag embedding methods outperform traditional semantic analysis methods. This solution allows social tags to be represented in dimensional emotion space without being limited to emotion corpus and quantify tags to facilitate emotion annotation.

7.2 Future Works

With the advance of technology in related domains, there is still a possible chance to improve music emotion recognition and social tags analysis, such as data collection, data processing and model design.

7.2.1 Data Collection

Combined with the result analysis in two MER scenarios, the dataset plays a significant performance role. The effect of the dataset is mainly reflected in two aspects: what kind of music excerpts are selected; the quality of annotation. For `PMEmo` dataset, the music excerpts are chorus part of pop music, which limits itself to one genre and repetitive acoustic pattern. Due to this, the training pattern is more likely to lack generalization. In contrast, `AllMusic` dataset contains songs with various genres that enhance the ability of model prediction. On the other hand, training samples with imbalanced labels bring about learning bias during model training such as `EmoMusic` dataset. Additionally, the annotation methods are also worth paying more attention to. In future work, dataset quality could be improved by complying with the following rules: consistent annotation from crowd sources or subjects mainly based on emotional responses from music rather than other stimuli;

balanced training samples; audio data covering multiple genres; diversified music excerpts. Currently, the relatively small data scale limits the performance of neural network models, which usually require large amounts of data samples to learn better patterns like the ones used in computer vision and NLP research. To take advantage of deep learning, it is better to build up a large-scale MER dataset, including emotion labels of quality.

In future work, the practical method of annotating music emotion through tags representation should be further studied combined with subjective tests as validation. Based on social tags analysis, the quantified tags could be considered the resource of emotion annotation, especially for the regression problem. On the other hand, it is better to build up the large-scale emotion ratings from music rather than other stimuli as mentioned above, which would boost the approximation accuracy during tag vector transformation.

7.2.2 Data Processing

In deep learning models for music emotion recognition, raw audio signals and spectrogram are most commonly used as inputs. Convolutional neural networks usually receive fixed-size inputs due to data structure requirements. However, the music samples are provided with various duration. Thus, data padding could be adopted during preprocessing. In future research, more effective padding methods could be considered to minimize the impact on emotional expression. Some audio augmentation methodologies may be applied.

Regarding music segmentation, segments with different duration are transformed into spectrograms of the same size due to the limitation of the training model. Such transformation may not be appropriate for all granularities of segments. Then how to optimize segmentation and extract features for those different segments properly could be taken into account in future work.

To enhance model robustness, data masking is utilized in many research areas. But some related details still need further discussion. For example, based on time and frequency dimensions in music research, how to select the appropriate span of the

mask; how many masking bands are applied; which position is proper for masking; what value is filled for the masking part and so on. More experiments could be conducted in a future study to consolidate the masking function and improve the performance.

Although neural network models require fewer human-engineered features as model inputs, we still need to learn more about musical knowledge and musical phenomena combined with music psychology to know how music and emotion interact, thus enabling us to create more appropriate music features.

7.2.3 Model Design

Focus on the performance of each dimension in emotion space, the model could be designed for valence and arousal separately. Especially for valence, the experimental results were relatively low in either the classification or regression models. In future work, we need to explore more about the relationship between music and valence dimension, thus designing a model better fit for valence prediction.

With the fast development of deep learning technology in many research fields, some advanced model designs could be applied. In the multi-view model or the two-stage model, it is possible to replace CNN or LSTM modules with more efficient neural networks. For example, segments could be viewed as time-series nodes that constitute a graph. Then we could attempt to utilize Graph Neural Network (GNN) to predict emotion in future research. On the other hand, the alternative model design needs to balance the computing cost and the model benefit.

In my current MER research, audio music data is the only input source of the training models. In future work, multiple feature sources can be combined with appropriate multimodal design and fusion strategies. Apart from the traditional music feature analysis for audio and lyrics, model design for biological signal data may be paid more attention to since such data involves the emotional response more exactly.

In my experiments, the data scale is relatively small due to the cost of collecting high-quality annotations. But deep learning usually requires large-scale training

samples to explore the learning pattern in case of overfitting. Facing music samples with limited annotation, some model architectures could be used for reference to enhance the training model or augment the data scale. One effective method is transfer learning, where a large-scale dataset model is first well-trained. Then this model is applied to a small-scale music dataset for feature representation or emotion recognition. For data augmentation, besides manipulating data directly, Generative Adversarial Network (GAN) could be explored to generate more samples for emotion recognition.

Considering social tags analysis, some advanced deep learning models for text analysis could be leveraged to replace current tag embedding algorithms. Also, transfer learning based on a large-scale well-trained text corpus might be utilized to explore the relationship between tags. In future studies, how to calculate music emotions based on their tags and use them in MER research could be taken into account.

Appendix A

Warriner's list

Warriner's list (Warriner et al., 2013) provides norms of valence, arousal, and dominance for 13,915 English lemmas. The scale ranges from 1 to 9 complying with the human intuitive low-to-high scale. 1 means either lowest valence or lowest arousal while 9 means highest valence or arousal. 5 could be considered as neutral status. All emotion ratings could be found under "electronic supplementary material" * column of the online version of that article.

Table A.1 : The Valence-Arousal reference for some terms cited in this thesis

Term	**Valence**	**Arousal**
happy	8.47	6.05
angry	2.53	6.2
sad	2.1	3.49
relax	7.82	2.38
heartbroken	1.95	6.6
black	5.4	3.58
dark	5.08	4.09
aggressive	3.08	5.87

Note: the value of valence and arousal cited in the table is the mean ratings.

*https://link.springer.com/article/10.3758/s13428-012-0314-xSecESM1

Bibliography

Alías, F., Socoró, J. C., and Sevillano, X. (2016). A review of physical and perceptual feature extraction techniques for speech, music and environmental sounds.

Aljanaki, A., Wiering, F., and Veltkamp, R. C. (2015a). Emotion based segmentation of musical audio. In *Proceedings of the 16th International Society for Music Information Retrieval Conference, ISMIR 2015*, pages 770–776.

Aljanaki, A., Wiering, F., and Veltkamp, R. C. (2016). Studying emotion induced by music through a crowdsourcing game. *Information Processing and Management*, 52(1):115–128.

Aljanaki, A., Yang, Y. H., and Soleymani, M. (2015b). Emotion in Music task at MediaEval 2015. In *CEUR Workshop Proceedings*, volume 1436.

Aljanaki, A., Yang, Y. H., and Soleymani, M. (2017). Developing a benchmark for emotional analysis of music. *PLoS ONE*, 12(3).

Bălan, O., Moise, G., Petrescu, L., Moldoveanu, A., Leordeanu, M., and Moldoveanu, F. (2020). Emotion classification based on biophysical signals and machine learning techniques. *Symmetry*, 12(1):1–22.

Beedie, C. J., Terry, P. C., and Lane, A. M. (2005). Distinctions between emotion and mood.

Bertin-Mahieux, T., Ellis, D. P., Whitman, B., and Lamere, P. (2011). The million song dataset. In *Proceedings of the 12th International Society for Music Information Retrieval Conference, ISMIR 2011*, volume 2, pages 591–596.

Bhattacharya, A. and Kadambari, K. V. (2018). A Multimodal Approach towards Emotion Recognition of Music using Audio and Lyrical Content. *arXiv*.

Bhattarai, B. and Lee, J. (2019). Automatic music mood detection using transfer learning and multilayer perceptron. *International Journal of Fuzzy Logic and Intelligent Systems*, 19(2):88–96.

Bian, W., Wang, J., Zhuang, B., Yang, J., Wang, S., and Xiao, J. (2019). Audio-Based Music Classification with DenseNet and Data Augmentation. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 11672 LNAI, pages 56–65.

Bigand, E., Vieillard, S., Madurell, F., Marozeau, J., and Dacquet, A. (2005). Multidimensional scaling of emotional responses to music: The effect of musical expertise and of the duration of the excerpts. *Cognition and Emotion*, 19(8):1113–1139.

Bogdanov, D., Wack, N., Gómez, E., Gulati, S., Herrera, P., Mayor, O., Roma, G., Salamon, J., Zapata, J., and Serra, X. (2013). Essentia: An audio analysis library for music information retrieval. In *Proceedings of the 14th International Society for Music Information Retrieval Conference, ISMIR 2013*, pages 493–498. International Society for Music Information Retrieval (ISMIR).

Bordes, A., Chopra, S., and Weston, J. (2014). Question Answering with Subgraph Embeddings. In *Empirical Methods in Natural Language Processing(EMNLP)*.

Cabrera, D., Ferguson, S., and Schubert, E. (2007). ' Psysound3 ': Software for Acoustical and Psychoacoustical Analysis of Sound Recordings. In *Proceedings of the 13th International Conference on Auditory Display*, page 356–363.

Çano, E. and Morisio, M. (2017a). Crowdsourcing Emotions in Music Domain. *International Journal of Artificial Intelligence & Applications*, 8(4):25–40.

Çano, E. and Morisio, M. (2017b). Music Mood Dataset Creation Based on Last FM Tags. In *2017 International Conference on Artificial Intelligence and Applications, Vienna, Austria*, pages 15–26.

Chen, J., Ma, M., Zheng, R., and Huang, L. (2021). SpecRec: An alternative solution for improving end-to-end speech-to-text translation via spectrogram re-

construction. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, volume 5, pages 3986–3990.

Chen, Y. A., Yang, Y. H., Wang, J. C., and Chen, H. (2015). The AMG1608 dataset for music emotion recognition. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, volume 2015-Augus, pages 693–697.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, pages 1724–1734.

Choi, K., Fazekas, G., and Sandler, M. (2016). Automatic tagging using deep convolutional neural networks. In *Proceedings of the 17th International Society for Music Information Retrieval Conference, ISMIR 2016*, pages 805–811.

Choi, K., Fazekas, G., Sandler, M., and Cho, K. (2017). Convolutional recurrent neural networks for music classification. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pages 2392–2396.

Chung, S. Y. and Yoon, H. J. (2012). Affective classification using Bayesian classifier and supervised learning. In *International Conference on Control, Automation and Systems*, pages 1768–1771.

Corona, H. and O'Mahony, M. P. (2015). An exploration of mood classification in the million songs dataset. In *Proceedings of the 12th International Conference in Sound and Music Computing, SMC 2015*, pages 363–370.

Cortes, C. and Vapnik, V. (1995). Support-Vector Networks. *Machine Learning*, 20(3):273–297.

Coutinho, E., Trigeorgis, G., Zafeiriou, S., and Schuller, B. (2015). Automatically estimating emotion in music with deep long-short term memory recurrent neural networks. In *CEUR Workshop Proceedings*, volume 1436.

de Berardinis, J., Cangelosi, A., and Coutinho, E. (2020). The multiple voices of musical emotions: source separation for improving music emotion recognition models and their interpretability. In *Proceedings of the 21st International Society for Music Information Retrieval Conference*, pages 310–217.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

Défossez, A., Usunier, N., Bottou, L., and Bach, F. (2019). Music Source Separation in the Waveform Domain. *arXiv*.

Delbouys, R., Hennequin, R., Piccoli, F., Royo-Letelier, J., and Moussallam, M. (2018). Music mood detection based on audio and lyrics with deep neural net. In *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018*, pages 370–375.

Deng, J. J. and Leung, C. H. (2015). Dynamic time warping for music retrieval using time series modeling of musical emotions. *IEEE Transactions on Affective Computing*, 6(2):137–151.

Devlin, J., Chang, M. W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, volume 1, pages 4171–4186.

Dieleman, S., Brakel, P., and Schrauwen, B. (2011). Audio-based music classification with a pretrained convolutional network. In *Proceedings of the 12th International Society for Music Information Retrieval Conference, ISMIR 2011*, pages 669–674.

Dong, Y., Yang, X., Zhao, X., and Li, J. (2019). Bidirectional Convolutional Recurrent Sparse Network (BCRSN): An Efficient Model for Music Emotion Recognition. *IEEE Transactions on Multimedia*, 21(12):3150–3163.

Eerola, T. and Vuoskoski, J. K. (2011). A comparison of the discrete and dimensional models of emotion in music. *Psychology of Music*, 39(1):18–49.

Evangelopoulos, N. E. (2013). Latent semantic analysis. *Wiley Interdisciplinary Reviews: Cognitive Science*, 4(6):683–692.

Eyben, F., Weninger, F., Gross, F., and Schuller, B. (2013). Recent developments in openSMILE, the munich open-source multimedia feature extractor. In *MM 2013 - Proceedings of the 2013 ACM Multimedia Conference*, pages 835–838.

Fan, J., Yang, Y. H., Dong, K., and Pasquier, P. (2020). A Comparative Study of Western and Chinese Classical Music Based on Soundscape Models. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, volume 2020-May, pages 521–525.

Fu, C., Liu, C., Ishi, C. T., and Ishiguro, H. (2020). Multi-modality emotion recognition model with gat-based multi-head inter-modality attention. *Sensors (Switzerland)*, 20(17):1–15.

Gabrielsson, A. and Lindström, E. (2001). The influence of musical structure on emotional expression. In *Music and emotion: Theory and research*, pages 223–248.

Ganguly, D., Roy, D., Mitra, M., and Jones, G. J. (2015). Word Embedding based Generalized Language Model for Information Retrieval. pages 795–798.

Giatsoglou, M., Vozalis, M. G., Diamantaras, K., Vakali, A., Sarigiannidis, G., and Chatzisavvas, K. C. (2017). Sentiment analysis leveraging emotions and word embeddings. *Expert Systems with Applications*, 69:214–224.

Girshick, R. (2015). Fast R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 2015 Inter, pages 1440–1448.

Gower, J. C. (2015). *Procrustes Analysis*.

Graves, A. and Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. In *Neural Networks*, volume 18, pages 602–610.

Grekow, J. (2017). Audio features dedicated to the detection of arousal and valence in music recordings. In *Proceedings - 2017 IEEE International Conference on INnovations in Intelligent SysTems and Applications, INISTA 2017*, pages 40–44.

Grekow, J. (2018). Music emotion maps in the arousal-valence space. In *Studies in Computational Intelligence*, volume 747, pages 95–106.

Hall, M. A. (2000). Feature Selection for Discrete and Numeric Class Machine Learning 1 Introduction. *Machine Learning Proc Seventeenth International conference on Machine Learning*, pages 1–16.

Han, B. J., Rho, S., Jun, S., and Hwang, E. (2010). Music emotion classification and context-based music recommendation. *Multimedia Tools and Applications*, 47(3):433–460.

He, K. and Sun, J. (2015). Convolutional neural networks at constrained time cost. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 07-12-June, pages 5353–5360.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 2015 Inter, pages 1026–1034.

He, N. and Ferguson, S. (2020a). Music Social Tags Representation in Dimensional Emotion Models. In *2020 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*, pages 819–826. IEEE.

He, N. and Ferguson, S. (2022). Music emotion recognition based on segment-level two-stage learning. *International Journal of Multimedia Information Retrieval*, 11(3):383–394.

He, N. A. and Ferguson, S. (2020b). Multi-view Neural Networks for Raw Audio-based Music Emotion Recognition. In *Proceedings - 2020 IEEE International Symposium on Multimedia, ISM 2020*.

Hershey, S., Chaudhuri, S., Ellis, D. P., Gemmeke, J. F., Jansen, A., Moore, R. C., Plakal, M., Platt, D., Saurous, R. A., Seybold, B., Slaney, M., Weiss, R. J., and Wilson, K. (2017). CNN architectures for large-scale audio classification. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pages 131–135.

Hevner, K. (1936). Experimental Studies of the Elements of Expression in Music. *The American Journal of Psychology*, 48(2):246.

Hochreiter, S. and Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, 12(1).

Hsieh, C. H., Li, Y. S., Hwang, B. J., and Hsiao, C. H. (2020). Detection of atrial fibrillation using 1D convolutional neural network. *Sensors (Switzerland)*, 20(7).

Hu, X., Choi, K., and Downie, J. S. (2017). A framework for evaluating multimodal music mood classification. *Journal of the Association for Information Science and Technology*, 68(2):273–285.

Hu, X. and Downie, J. S. (2007). Exploring mood metadata: Relationships with genre, artist and usage metadata. In *Proceedings of the 8th International Conference on Music Information Retrieval, ISMIR 2007*, pages 67–72.

Hu, X., Downie, J. S., and Ehmann, A. F. (2009). Lyric text mining in music mood classification. *American Music*, 183(5,049):2–209.

Hu, X., Downie, J. S., Laurier, C., Bay, M., and Ehmann, A. F. (2008). The 2007 mirex audio mood classification task: Lessons learned. In *ISMIR 2008 - 9th International Conference on Music Information Retrieval*, pages 462–467.

Huang, M., Rong, W., Arjannikov, T., Jiang, N., and Xiong, Z. (2016). Bi-modal deep boltzmann machine based musical emotion classification. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 9887 LNCS, pages 199–207.

Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *32nd International Conference on Machine Learning, ICML 2015*, volume 1, pages 448–456.

Jeon, B., Kim, C., Kim, A., Kim, D., Park, J., and Ha, J. W. (2017). Music emotion recognition via end-To-end multimodal neural networks. In *CEUR Workshop Proceedings*, volume 1905.

Jeong, I.-Y. and Lee, K. (2016). Learning temporal features using a deep neural network and its application to music genre classification. In *Proceedings of the 17th International Society for Music Information Retrieval Conference*, pages 434–440.

Kallinen, K. and Ravaja, N. (2006). Emotion perceived and emotion felt: Same and different. *Musicae Scientiae*, 10(2):191–213.

Kartikay, A., Ganesan, H., and Ladwani, V. M. (2016). Classification of Music into Moods using Musical Features. In *Proceedings of the International Conference on Inventive Computation Technologies, ICICT 2016*, volume 2016.

Ketai, R. (1975). Affect, mood, emotion, and feeling: semantic considerations. *American Journal of Psychiatry*, 132(11):1215–1217.

Kim, Y. E., Schmidt, E., and Emelle, L. (2008). MoodSwings: A collaborative game for music mood label collection. In *ISMIR 2008 - 9th International Conference on Music Information Retrieval*, pages 231–236.

Kingma, D. P. and Ba, J. L. (2015). Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.

Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M., and Inman, D. J. (2021). 1D convolutional neural networks and applications: A survey. *Mechanical Systems and Signal Processing*, 151.

Knautz, K., Neal, D. R., Schmidt, S., Siebenlist, T., and Stock, W. G. (2011). Finding Emotional-Laden Resources on the World Wide Web.

Koelstra, S., Mühl, C., Soleymani, M., Lee, J. S., Yazdani, A., Ebrahimi, T., Pun, T., Nijholt, A., and Patras, I. (2012). DEAP: A database for emotion analysis; Using physiological signals. *IEEE Transactions on Affective Computing*, 3(1):18–31.

Kraemer, G., Reichstein, M., and Mahecha, M. D. (2018). dimRed and coRanking-unifying dimensionality reduction in R. *R Journal*, 10(1):342–358.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90.

Kruskal, J. B. (1964). Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27.

Lamere, P. (2008). Social tagging and music information retrieval. *Journal of New Music Research*, 37(2):101–114.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., and Dyer, C. (2016). Neural architectures for named entity recognition. *2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2016 - Proceedings of the Conference*, pages 260–270.

Lartillot, O., Toiviainen, P., and Eerola, T. (2008). A matlab toolbox for music information retrieval. In *Studies in Classification, Data Analysis, and Knowledge Organization*, pages 261–268.

Laurier, C., Meyers, O., Serrà, J., Blech, M., Herrera, P., and Serra, X. (2010).

Indexing music by mood: Design and integration of an automatic content-based annotator. In *Multimedia Tools and Applications*, volume 48, pages 161–184.

Laurier, C., Sordo, M., and Serrà, J. (2009). Music mood representations from social tags. In *Proc. International Society for Music Information Retrieval Conference*, pages 381–386.

Law, E. L., Ahn, L. V., Dannenberg, R. B., and Crawford, M. (2007). Tagatune: A game for music and sound annotation. In *Proceedings of the 8th International Conference on Music Information Retrieval, ISMIR 2007*, pages 361–364.

Lee, J., Park, J., Kim, K. L., and Nam, J. (2017). Sample-level deep convolutional neural networks for music auto-tagging using raw waveforms. In *Proceedings of the 14th Sound and Music Computing Conference 2017, SMC 2017*, pages 220–226.

Lee, J., Park, J., Kim, K. L., and Nam, J. (2018). SampleCNN: End-to-end deep convolutional neural networks using very small filters for music classification. *Applied Sciences (Switzerland)*, 8(1).

Levy, M. and Sandler, M. (2008). Learning latent semantic models for music from social tags. *Journal of New Music Research*, 37(2):137–150.

Levy, M. and Sandler, M. B. (2007). A semantic space for music derived from social tags. *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR 2007)*.

Li, J., Gao, S., Han, N., Fang, Z., and Liao, J. (2015). Music Mood Classification via Deep Belief Network. In *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, pages 1241–1245.

Li, X., Tian, J., Xu, M., Ning, Y., and Cai, L. (2016). DBLSTM-based multi-scale fusion for dynamic emotion prediction in music. In *Proceedings - IEEE International Conference on Multimedia and Expo*, volume 2016-Augus.

Li, Y., Yang, M., and Zhang, Z. (2019). A Survey of Multi-View Representation

Learning. *IEEE Transactions on Knowledge and Data Engineering*, 31(10):1863–1883.

Lian, Z., Li, Y., Tao, J., and Huang, J. (2018). Investigation of Multimodal Features, Classifiers and Fusion Methods for Emotion Recognition.

Lidy, T. and Schindler, A. (2016). Parallel Convolutional Neural Networks for Music Genre and Mood Classification. Technical report.

Lin, C., Liu, M., Hsiung, W., and Jhang, J. (2016). Music emotion recognition based on two-level support vector classification. In *2016 International Conference on Machine Learning and Cybernetics (ICMLC)*, pages 375–389.

Lin, Y.-C., Yang, Y.-H., and Chen, H. H. (2011). Exploiting online music tags for music emotion classification. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 7S(1):1–16.

Lin, Y. C., Yang, Y. H., Chen, H. H., Liao, I. B., and Ho, Y. C. (2009). Exploiting genre for music emotion classification. In *Proceedings - 2009 IEEE International Conference on Multimedia and Expo, ICME 2009*, pages 618–621.

Lin, Y. P., Wang, C. H., Jung, T. P., Wu, T. L., Jeng, S. K., Duann, J. R., and Chen, J. H. (2010). EEG-based emotion recognition in music listening. *IEEE Transactions on Biomedical Engineering*, 57(7):1798–1806.

Liu, A. T., Yang, S. W., Chi, P. H., Hsu, P. C., and Lee, H. Y. (2020). Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, volume 2020-May, pages 6419–6423.

Madiraju, N. S., Sadat, S. M., Fisher, D., and Karimabadi, H. (2018). Deep temporal clustering: Fully unsupervised learning of time-domain features.

Makris, D., Karydis, I., and Sioutas, S. (2015). The Greek Music Dataset. pages 1–7.

Malik, M., Adavanne, S., Drossos, K., Virtanen, T., Ticha, D., and Jarina, R. (2017). Stacked convolutional and recurrent neural networks for music emotion recognition. In *Proceedings of the 14th Sound and Music Computing Conference 2017, SMC 2017*, pages 208–213.

McFee, B., Raffel, C., Liang, D., Ellis, D., McVicar, M., Battenberg, E., and Nieto, O. (2015). librosa: Audio and Music Signal Analysis in Python. In *Proceedings of the 14th Python in Science Conference*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Munezero, M., Montero, C. S., Sutinen, E., and Pajunen, J. (2014). Are they different? affect, feeling, emotion, sentiment, and opinion detection in text. *IEEE Transactions on Affective Computing*, 5(2):101–111.

Nawaz, R., Nisar, H., Voon, Y. V., and Yee, T. P. (2018). Acoustic feature extraction from music songs to predict emotions using neural networks. In *2nd International Conference on BioSignal Analysis, Processing and Systems, ICBAPS 2018*, pages 166–170. Institute of Electrical and Electronics Engineers Inc.

Nayal, J. S., Joshi, A., and Kumar, B. (2019). Emotion recognition in songs via Bayesian deep learning. In *ACM International Conference Proceeding Series*, pages 235–238. Association for Computing Machinery.

Nordström, H. and Laukka, P. (2019). The time course of emotion recognition in speech and music. *The Journal of the Acoustical Society of America*, 145(5):3058–3074.

Orjesek, R., Jarina, R., Chmulik, M., and Kuba, M. (2019). DNN based music emotion recognition from raw audio signal. In *2019 29th International Conference Radioelektronika, RADIOELEKTRONIKA 2019 - Microwave and Radio Electronics Week, MAREW 2019*, pages 1–4. IEEE.

Palaz, D., Magimai.-Doss, M., and Collobert, R. (2015). Convolutional Neural Networks-based continuous speech recognition using raw speech signal. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, volume 2015-Augus, pages 4295–4299.

Paltoglou, G. and Thelwall, M. (2013). Seeing stars of valence and arousal in blog posts. *IEEE Transactions on Affective Computing*, 4(1):116–123.

Panda, R., Malheiro, R. M., and Paiva, R. P. (2018). Novel audio features for music emotion recognition. *IEEE Transactions on Affective Computing*.

Panda, R., Rocha, B., and Paiva, R. P. (2015). Music emotion recognition with standard and melodic audio features. *Applied Artificial Intelligence*, 29(4):313–334.

Park, D. S., Chan, W., Zhang, Y., Chiu, C. C., Zoph, B., Cubuk, E. D., and Le, Q. V. (2019). Specaugment: A simple data augmentation method for automatic speech recognition. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, volume 2019-Septe, pages 2613–2617.

Pascanu, R., Gulcehre, C., Cho, K., and Bengio, Y. (2014). How to construct deep recurrent neural networks. In *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*.

Patra, B. G., Das, D., and Bandyopadhyay, S. (2016). Multimodal mood classification framework for Hindi songs. *Computacion y Sistemas*, 20(3):515–526.

Pegoraro Santana, I. A., Pinhelli, F., Donini, J., Catharin, L., Mangolin, R. B., Da Costa, Y. M. E., Delisandra Feltrim, V., and Domingues, M. A. (2020). Music4All: A New Music Database and Its Applications. In *International Conference on Systems, Signals, and Image Processing*, volume 2020-July, pages 399–404.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Poria, S., Cambria, E., Howard, N., Huang, G. B., and Hussain, A. (2016). Fusing audio, visual and textual clues for sentiment analysis from multimodal content. *Neurocomputing*, 174:50–59.

Purwins, H., Li, B., Virtanen, T., Schlüter, J., Chang, S. Y., and Sainath, T. (2019). Deep Learning for Audio Signal Processing. *IEEE Journal on Selected Topics in Signal Processing*, 13(2):206–219.

Robnik-Šikonja, M. and Kononenko, I. (2003). Theoretical and Empirical Analysis of ReliefF and RReliefF. *Machine Learning*, 53(1-2):23–69.

Rozgic, V., Vitaladevuni, S. N., and Prasad, R. (2013). Robust EEG emotion classification using segment level decision fusion. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pages 1286–1290.

Russell, J. A. (1980). A circumplex model of affect. *Journal of Personality and Social Psychology*, 39(6):1161–1178.

Russell, J. A. and Barrett, L. F. (1999). Core affect, prototypical emotional episodes, and other things called emotion: Dissecting the elephant. *Journal of Personality and Social Psychology*, 76(5):805–819.

Saari, P. and Eerola, T. (2014). Semantic Computing of Moods Based on Tags in Social Media of Music. *IEEE Transactions on Knowledge and Data Engineering*.

Saari, P., Eerola, T., and Lartillot, O. (2011). Generalizability and Simplicity as Criteria in Feature Selection: Application to Mood Classification in Music. *IEEE Transactions on Audio, Speech and Language Processing*, 19(6):1802–1812.

Salton, G., Wong, A., and Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620.

Sarkar, R., Choudhury, S., Dutta, S., Roy, A., and Saha, S. K. (2020). Recognition of emotion in music based on deep convolutional neural network. *Multimedia Tools and Applications*, 79(1-2):765–783.

Scherer, K. R. (2005). What are emotions? and how can they be measured? *Social Science Information*, 44(4):695–729.

Schimmack, U. and Grob, A. (2000). Dimensional models of core affect: A quantitative comparison by means of structural equation modeling. *European Journal of Personality*, 14(4):325–345.

Schindler, A. and Knees, P. (2019). Multi-Task Music Representation Learning from Multi-Label Embeddings. In *Proceedings - International Workshop on Content-Based Multimedia Indexing*, volume 2019-Septe, pages 1–6. IEEE.

Schmidt, E. M., Turnbull, D., and Kim, Y. E. (2010). Feature selection for content-based, time-varying musical emotion regression. In *MIR 2010 - Proceedings of the 2010 ACM SIGMM International Conference on Multimedia Information Retrieval*, pages 267–273.

Schölkopf, B., Smola, A. J., Williamson, R. C., and Bartlett, P. L. (2000). New support vector algorithms. *Neural Computation*, 12(5):1207–1245.

Senac, C., Pellegrini, T., Mouret, F., and Pinquier, J. (2017). Music feature maps with convolutional neural networks for music genre classification. In *ACM International Conference Proceeding Series*, volume Part F1301, page 19. ACM.

Sharma, H., Gupta, S., Sharma, Y., and Purwar, A. (2020). A New Model for Emotion Prediction in Music. In *2020 6th International Conference on Signal Processing and Communication, ICSC 2020*, pages 156–161.

Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.

Socher, R., Bauer, J., Manning, C. D., and Ng, A. Y. (2013). Parsing With Compositional Vector Grammars. *Advances in Neural Information Processing Systems*, pages 455–465.

Soleymani, M., Asghari-Esfeden, S., Fu, Y., and Pantic, M. (2016). Analysis of EEG Signals and Facial Expressions for Continuous Emotion Detection. *IEEE Transactions on Affective Computing*, 7(1):17–28.

Soleymani, M., Caro, M. N., Schmidt, E. M., Sha, C. Y., and Yang, Y. H. (2013). 1000 Songs for Emotional Analysis of Music. In *CrowdMM 2013 - Proceedings of the 2nd ACM International Workshop on Crowdsourcing for Multimedia*, pages 1–6.

Soleymani, M., Pantic, M., and Pun, T. (2012). Multimodal emotion recognition in response to videos. *IEEE Transactions on Affective Computing*, 3(2):211–223.

Solomatine, D. P. and Shrestha, D. L. (2004). AdaBoost.RT: A boosting algorithm for regression problems. In *IEEE International Conference on Neural Networks - Conference Proceedings*, volume 2, pages 1163–1168.

Song, Y., Dixon, S., and Pearce, M. (2012). Evaluation of musical features for emotion classification. In *Proceedings of the 13th International Society for Music Information Retrieval Conference, ISMIR 2012*, pages 523–528.

Song, Y., Dixon, S., Pearce, M., and Halpern, A. (2013). Do Online Social Tags Predict Percieved or Induced Emotial Responses to Music? *In Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR 2013)*, pages 89–94.

Song, Y., Dixon, S., Pearce, M. T., and Halpern, A. R. (2016). Perceived and Induced Emotion Responses to Popular Music: Categorical and Dimensional Models. *Music Perception: An Interdisciplinary Journal*, 33(4):472–492.

Srivastava, N. and Salakhutdinov, R. (2014). Multimodal learning with Deep Boltzmann Machines. *Journal of Machine Learning Research*, 15:2949–2980.

Su, H., Maji, S., Kalogerakis, E., and Learned-Miller, E. (2015). Multi-view convolutional neural networks for 3D shape recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 2015 Inter, pages 945–953.

Thayer, R. (1989). *The biopsychology of mood and arousal*.

Tibshirani, R. (2011). Regression shrinkage and selection via the lasso: A retrospective. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, 73(3).

Tripathi, S., Acharya, S., Sharma, R., Mittal, S., and Bhattacharya, S. (2017). Using Deep and Convolutional Neural Networks for Accurate Emotion Classification on DEAP Dataset. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, volume 174, pages 4746–4752.

Turian, J., Ratinov, L., and Bengio, Y. (2010). Word representations : A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL-10)*, pages 384–394.

Turnbull, D., Barrington, L., Torres, D., and Lanckriet, G. (2008). Semantic annotation and retrieval of music and sound effects. *IEEE Transactions on Audio, Speech and Language Processing*, 16(2):467–476.

Tzanetakis, G. and Cook, P. (2000). MARSYAS: A framework for audio analysis. *Organised Sound*, 4(3):169–175.

Vale, P. M. F. (2017). *The role of artist and genre on music emotion recognition*. PhD thesis, Universidade Nova de Lisboa.

Vatolkin, I. and Nagathil, A. (2019). Evaluation of Audio Feature Groups for the Prediction of Arousal and Valence in Music. In *Studies in Classification, Data Analysis, and Knowledge Organization*, pages 305–326. Springer.

Verma, G. K. and Tiwary, U. S. (2017). Affect representation and recognition in 3D continuous valence–arousal–dominance space. *Multimedia Tools and Applications*, 76(2).

Wang, Q., Su, F., and Wang, Y. (2019). A hierarchical attentive deep neural network model for semantic music annotation integrating multiple music representations.

In *ICMR 2019 - Proceedings of the 2019 ACM International Conference on Multimedia Retrieval*, pages 150–158. Association for Computing Machinery, Inc.

Wang, W., Tang, Q., and Livescu, K. (2020). Unsupervised Pre-Training of Bidirectional Speech Encoders via Masked Reconstruction. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, volume 2020-May, pages 6889–6893.

Warriner, A. B., Kuperman, V., and Brysbaert, M. (2013). Norms of valence, arousal, and dominance for 13,915 English lemmas. *Behavior Research Methods*, 45(4):1191–1207.

Wen, Q., Sun, L., Song, X., Gao, J., Wang, X., and Xu, H. (2020). Time Series Data Augmentation for Deep Learning: A Survey. *arXiv preprint arXiv:2002.12478*.

Weninger, F., Eyben, F., and Schuller, B. (2014). On-line continuous-time music mood regression with deep recurrent neural networks. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*.

Wu, B., Zhong, E., Horner, A., and Yang, Q. (2014). Music emotion recognition by multi-label multi-layer multi-instance multi-view learning. In *MM 2014 - Proceedings of the 2014 ACM Conference on Multimedia*, pages 117–126.

Wu, B., Zhong, E., Hu, D. H., Horner, A., and Yang, Q. (2013). SMART: Semi-supervised music emotion recognition with social tagging. In *Proceedings of the 2013 SIAM International Conference on Data Mining, SDM 2013*, pages 279–287.

Wu, H. C., Luk, R. W. P., Wong, K. F., and Kwok, K. L. (2008). Interpreting TF-IDF term weights as making relevance decisions. *ACM Transactions on Information Systems*, 26(3):1–37.

Xianyu, H., Xu, M., Wu, Z., and Cai, L. (2016). Heterogeneity-entropy based unsupervised feature learning for personality prediction with cross-media data. In *Proceedings - IEEE International Conference on Multimedia and Expo*, volume 2016-Augus.

Xiao, Z., Dellandrea, E., Dou, W., and Chen, L. (2008). What is the best segment duration for music mood analysis ? In *2008 International Workshop on Content-Based Multimedia Indexing, CBMI 2008, Conference Proceedings*, pages 17–24.

Yang, Y. H. and Chen, H. H. (2011). Ranking-based emotion recognition for music organization and retrieval. *IEEE Transactions on Audio, Speech and Language Processing*, 19(4):762–774.

Yang, Y.-H. and Chen, H. H. (2012). Machine Recognition of Music Emotion: A Review. *ACM Transactions on Intelligent Systems and Technology*, 3(3):1–30.

Yang, Y. H., Lin, Y. C., Su, Y. F., and Chen, H. H. (2007). Music emotion classification: A regression approach. In *Proceedings of the 2007 IEEE International Conference on Multimedia and Expo, ICME 2007*, pages 208–211.

Yang, Y. H., Lin, Y. C., Su, Y. F., and Chen, H. H. (2008). A regression approach to music emotion recognition. *IEEE Transactions on Audio, Speech and Language Processing*, 16(2):448–457.

Yeh, C. H., Tseng, W. Y., Chen, C. Y., Lin, Y. D., Tsai, Y. R., Bi, H. I., Lin, Y. C., and Lin, H. Y. (2014). Popular music representation: chorus detection & emotion recognition. *Multimedia Tools and Applications*, 73(3):2103–2128.

Yin, G., Sun, S., Yu, D., and Zhang, K. (2020). A Efficient Multimodal Framework for Large Scale Emotion Recognition by Fusing Music and Electrodermal Activity Signals.

Yin, G., Sun, S., Zhang, H., Yu, D., Li, C., Zhang, K., and Zou, N. (2019). User Independent Emotion Recognition with Residual Signal-Image Network. In *Proceedings - International Conference on Image Processing, ICIP*, pages 3277–3281.

Yin, Z., Wang, Y., Liu, L., Zhang, W., and Zhang, J. (2017). Cross-subject EEG feature selection for emotion recognition using transfer recursive feature elimination. *Frontiers in Neurorobotics*, 11(APR).

Zahid, M. U., Kiranyaz, S., Ince, T., Devecioglu, O. C., Chowdhury, M. E. H., Khandakar, A., Tahir, A., and Gabbouj, M. (2021). Robust R-Peak Detection in Low-Quality Holter ECGs using 1D Convolutional Neural Network. *IEEE Transactions on Biomedical Engineering*.

Zhang, J., Yin, Z., Chen, P., and Nichele, S. (2020). Emotion recognition using multi-modal data and machine learning techniques: A tutorial and review. *Information Fusion*, 59:103–126.

Zhang, J. L., Huang, X. L., Yang, L. F., Xu, Y., and Sun, S. T. (2017). Feature selection and feature learning in arousal dimension of music emotion by using shrinkage methods. *Multimedia Systems*, 23(2):251–264.

Zhang, K., Zhang, H., Li, S., Yang, C., and Sun, L. (2018). The PMEmo dataset for music emotion recognition. In *ICMR 2018 - Proceedings of the 2018 ACM International Conference on Multimedia Retrieval*, pages 135–142.

Zhang, P., Zheng, X., Zhang, W., Li, S., Qian, S., He, W., Zhang, S., and Wang, Z. (2015). A deep neural network for modeling music. In *ICMR 2015 - Proceedings of the 2015 ACM International Conference on Multimedia Retrieval*, pages 379–386.

Zhao, S., Wang, S., Soleymani, M., Joshi, D., and Ji, Q. (2019). Affective computing for large-scale heterogeneous multimedia data: A survey.

Zhao, Y. and Guo, J. (2021). MusiCoder: A Universal Music-Acoustic Encoder Based on Transformer.

Zhong, P., Wang, D., and Miao, C. (2020). EEG-Based Emotion Recognition Using Regularized Graph Neural Networks. *IEEE Transactions on Affective Computing*, pages 1–1.

Zhou, J., Chen, X., and Yang, D. (2019). Multimodel music emotion recognition using unsupervised deep neural networks. In *Lecture Notes in Electrical Engineering*, volume 568, pages 27–39.