
# Music Emotion Recognition Using Machine Learning to Analyze Lyrics and Audio Features

## B.Sc. in Artificial Intelligence and Computer Science

**Student Name:** Yuchen Zhu

**Student ID:** 2335100

**Supervisor Name:** Dr. Jizheng Wan

**Academic Year:** 2023/2024

**Word Count:**

# Abstract

This research is dedicated to creating a comprehensive music emotion analysis model, aimed at unraveling the complex emotional content in musical compositions through the analysis of lyrics and audio data. A pivotal aspect of the research was the replication and enhancement of methods from a key paper that utilized the MoodyLyrics dataset. This enhancement includes combining BoW, Tf-idf, and Word2Vec text embedding techniques that focus more on word frequency and local context with advanced preprocessing methods and machine learning models such as SVM, CNN, and NB. A notable achievement was the SVM+Tf-idf combination, which attained a 94% F1 score in single lyric analysis, demonstrating significant progress. Furthermore, the research found that incorporating Spotify audio features significantly boosts the model's performance and generalizability. This was evidenced by the performance of a CNN+Densenet+Word2Vec on the MoodyLyrics4Q dataset, where it surpassed the baseline with a 68% F1 score. Additionally, the research applied the model to predict the emotional content of Spotify's Top 100 songs from 2013 to 2023, validating these predictions with real-world societal data to provide an accurate dataset for future research. This research not only demonstrates that combining lyrics and audio data can enhance understanding of musical emotion, but also provides an important resource for in-depth exploration of the emotional landscape in contemporary music.

# Acknowledgements

I want to express my deepest gratitude to my supervisor, Dr. Jizheng Wan, for his invaluable insights and guidance throughout this research. I am also thankful to my inspector, Dr. Masoumeh Mansouri, for her valuable feedback during the project proposal and demonstration stages. Additionally, I want to thank the University of Birmingham as an institution, and to all lecturers in the B.Sc. program in Artificial Intelligence and Computer Science, who have provided us with quality education and the necessary resources to complete this research. This work would not have been possible without their imparted academic knowledge. Finally, I owe immense gratitude to my parents, family, and friends for their unwavering encouragement and support. Without them, none of this would have been achievable.

# Acronyms

**CNN** = Convolutional neural network

**NB** = Naive Bayes

**LSTM** = Long Short-Term Memory

**BI-LSTM** = Bidirectional Long Short-Term Memory

**KNN** = K-Nearest Neighbors

**SVM** = Support Vector Machine

**RF** = Random Forest

**Lemma** = Lemmatization

**Stem** = Stemming

**LR** = Lowercase Conversion

**NR** = Noise Removal

**SR** = Stopword Removal

**PCA** = Principal Component Analysis

**t-SNE** = t-distributed Stochastic Neighbor Embedding

# Contents

# List of Figures

# List of Tables

CHAPTER 1

# Introduction

## 1.1 Background and Motivation

Music, an ancient and universal art form, has played a significant role in human history. It is capable of containing and expressing emotions, a relationship that has been explored by various theories [1][2]. Contemporary empirical studies have further supported this connection, identifying three types of musical emotions: embodied, cognitive, and associative[3]. Recent research has discovered that, regardless of individual musical preferences, the structure of music can trigger specific emotions[4]. These findings collectively emphasize the powerful impact of music on our emotional experiences. Whether in moments of celebration or mourning, music uniquely touches our souls, awakening deep feelings of joy, sadness, love, or nostalgia. Across different cultures and societies, music plays a central role, serving not just as a medium for emotional expression, but also as a bridge connecting people through emotions. Furthermore, from a neuroscience perspective, music has been shown to influence complex neurobiological processes. It can be used as an alternative therapy for various mental disorders[5], highlighting its potential as a tool for promoting mental health. In the past decade, with the exponential growth of easily accessible digital music libraries, the challenge of effectively organizing and searching music and its related data has become increasingly prominent. Music Information Retrieval (MIR), as a scientific field within this domain, is rapidly advancing towards automated systems for searching and organizing music and related data. While common search and retrieval categories like artists or genres can be quantified into a 'correct' (or generally agreed upon) answer, the emotional expression

inherent in the music itself can be highly subjective and difficult to quantify[6]. Therefore, the subjectivity and complexity of its emotional content mean that traditional methods may not be sufficient to meet growing needs[7]. Searching music by emotion is one of the main criteria used by users[8][9], hence real music databases from websites like Spotify and Last.fm are growing daily, requiring extensive manual work to stay updated. Lyrics and audio, as the two main components of music, are often used by artists to convey emotional dimensions, with different aspects potentially conveying vastly different emotional dimensions and intensities. In today's data-driven environment, the rapid development of machine learning and deep learning has opened new possibilities for analyzing complex emotional content in music. These advanced technologies allow us to automate the processing and analysis of large music datasets, revealing the hidden emotional layers in musical compositions, particularly when dealing with lyrics rich in emotional expression and complex audio features.

## 1.2 Issue and Existing Solutions

Despite the progress in music emotion analysis, existing approaches often face limitations in comprehensively capturing the emotional essence of music. Many studies focus predominantly on either lyrical content or audio features, overlooking the synergistic potential of integrating both. This oversight can lead to a partial understanding of music's emotional impact, limiting the effectiveness of applications in areas such as personalized music recommendation, therapeutic interventions, and emotional analysis in music education.

## 1.3 Objectives and Contributions

This project aims to apply machine learning, deep learning, and natural language processing technologies to develop a comprehensive model that analyzes lyrics and audio data to gain a deeper understanding of emotional expression in music. The goal is to surpass existing benchmarks for emotional classification and enhance the model's generalization ability. This model is expected not only to provide a new perspective for the analysis of musical emotions, but also to have a positive practical impact on fields such as music recommendation systems, emotional therapy, and music education, offering a more detailed and comprehensive approach to music emotion analysis.

In summary, the primary contributions of the project include the following:

1. Replicating an existing model from academic literature, the project laid a solid foundation for future enhancements, effectively establishing a baseline for the research approach and confirming its efficacy.

2. Optimizing the lyrics only model based on this replication, significant improvements were made in embedding techniques, preprocessing methods, and adjustments to the model's architecture and parameters, resulting in surpassing the performance baseline of the original research paper.

3. Incorporating audio features, the development of an Integrated Model was achieved, exhibiting increased robustness and generalizability and surpassing the performance benchmarks of subsequent studies in the field.

4. Utilizing the integrated model to predict and label emotions in top songs over the past decade, the project not only compiled a valuable dataset but also used it for assessing the model's effectiveness against real-world events, showcasing its capability in accurately reflecting emotional trends for music recommendation and emotional analysis.

## 1.4 Report Organization

This report is organized into seven chapters.Chapter 1 starts with a broad overview, introducing the background of the project, its specific implementation, and results. Chapter 2 presents the necessary background knowledge, including historical and technical contexts, as well as related work.Chapter 3 details the specific methodologies of the research and some visual results. Chapter 4 provides the quantified research results. Chapter 5 offers an in-depth analysis of the research findings and the proposed methods. Chapter 6 discusses future research directions and potential limitations of this research. Finally, the report concludes in Chapter 7.

# CHAPTER 2

# Literature Review

## 2.1 Music Emotion Recognition

In recent years, Music Emotion Recognition (MER) [6] has gradually become a hot topic of research and has emerged as one of the key areas in the field of Music Information Retrieval (MIR) [10]. The annual conferences of the International Society for Music Information Retrieval (ISMIR), which began in 2000, signify the maturation and development of the MIR field. MER aims to delve deeply into and accurately identify the emotions and moods expressed in musical works, achieving personalized music recommendations and precise categorization. The ability of music to express and evoke emotions remains somewhat enigmatic. Different schools of psychology and musicology have their interpretations and explanations for this phenomenon.The research by Patrik N. Juslin, Erik Lindström [11], and others reveal, on one hand, the psychological perspective of how individuals experience emotional responses induced by music. On the other hand, musicology primarily investigates how music, as an art form, is crafted and expressed emotionally.

When it comes to understanding the relationship between music and emotions, psychological models of emotions in music have been shown to be valuable tools. These models facilitate the reduction of the emotional spectrum into a set of practical categories. An early and groundbreaking study in this regard is Hevner's research from 1936 [12], which outlined a categorical model consisting of 66 emotional adjectives, grouped into eight categories(Figure 2.1). Although the basic form of this model is not widely used in MER, it has been a reference point in some studies

employing categorical models.

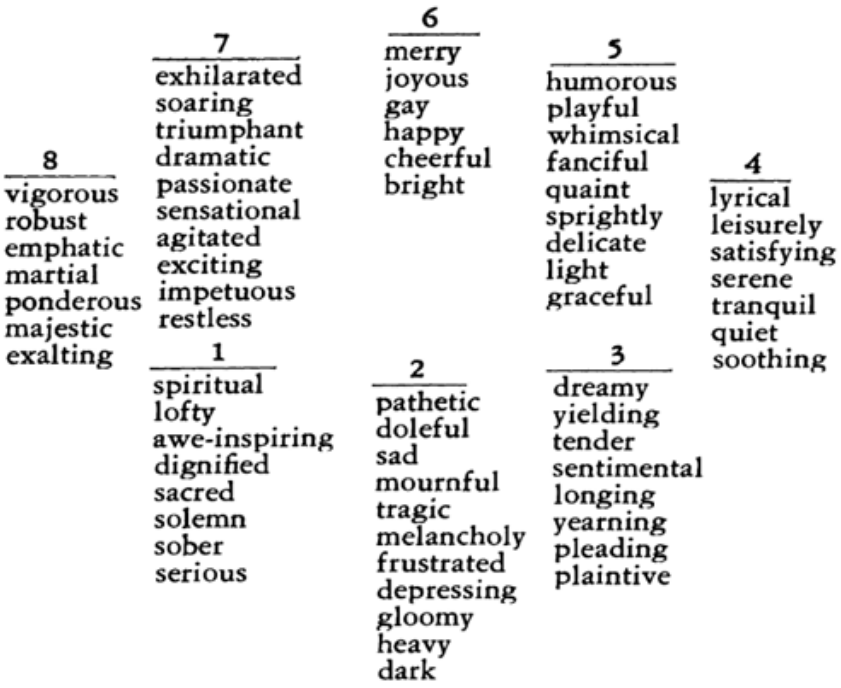


Figure 2.1: Hevner's Emotional Adjective Model

The two-dimensional model of emotion by Russell[13], which classifies and understands emotions through dimensions of valence (the positive or negative aspect of emotions) and arousal (the intensity of emotions), is more valued in MER(Figure 2.2).

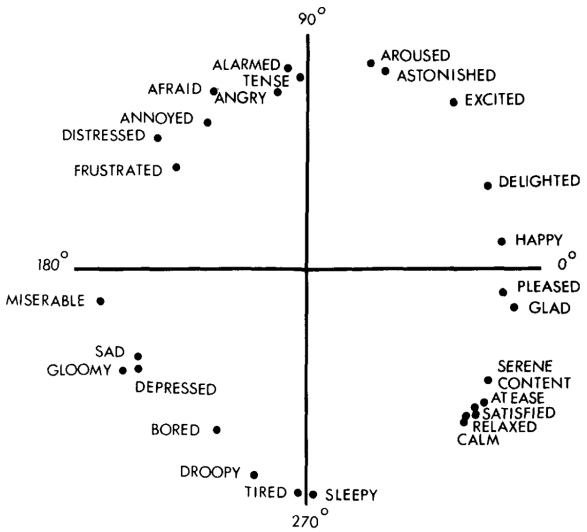


Figure 2.2: Russel Model

However, the complexity of musical emotions often surpasses the scope of this model. Leading researchers like Schimmack and Grob [14] have developed a three-dimensional emotional model,

which includes dimensions like pleasure-unpleasure, arousal-sleepiness, and tension-relaxation.

MER involves analyzing various features of music (such as melody, rhythm, harmony, lyrics, and background) to identify and categorize the emotional content in music. This deep exploration not only drives the rapid development of personalized music systems but also reveals significant social value and potential for commercial applications [15]. Studies conducted by Yang in 2008 [16] and 2014 [17] have demonstrated the application of automatic music emotion recognition technology in developing emotion-based music therapy methods. Y Liu and O Sourina [18] have also proposed a real-time EEG-based emotion recognition algorithm for music therapy, highlighting the significance and prospects of music emotion recognition.

## 2.2 Music Emotion Recognition Technology

With technological advancements, the information technology applied in the field of Music Emotion Recognition (MER) has become increasingly rich. Emotion analysis in music can be approached from many aspects. Studies show that the independence of lyrics and tune is consistent with the modular organization of the human cognitive system, as indicated by M. Besson, F. Faïta, I. Peretz, A. Bonnel, J. Requin in 1998[19]. Although music is a multimodal data form, as stated by Rudolf Mayer and Andreas Rauber[20], and might have emotional orientations influenced by the corresponding artist and cultural background, studies by Michael Fell and C. Sporleder[21] have proven that lyrics and audio are the mainstays of music analysis. Consequently, most research in this field focuses on analyzing music's lyrics and audio.

The corresponding emotional labels for lyrics and audio, and their acquisition, remain a challenge in this field due to copyright issues and the subjectivity of music, leading to a lack of rich datasets. In the lyrics aspect of MER, researchers Erion Çano and Maurizio Morisio[22] used ANEW and WordNet lexicons for word representation and clustering. They integrated the emotional values of each sentence and the entire song, creating a MoodyLyrics dataset organized into four quadrants based on Russell's model of emotions, available for public use. They employed dictionary and clustering methods and compared them with lyrics datasets annotated with user tags and human subjects, achieving an accuracy of 74.25%. This dataset focuses only on the emotional dimensions and labeling of lyric texts. In a study by Abdillah J et al. [23], the MoodyLyrics dataset was utilized, employing a combination of BiLSTM+GloVe with regularization layers and hyperparameter tuning. This approach achieved an accuracy of 91% on single lyric texts. However, this study did not consider the audio dimension.

Subsequently, Erion Çano created the MoodyLyrics4Q[24], a dataset based on the Russell

emotional model for classification standards and collected from the music community platform LastFM based on subjective user tags. This allowed for a broader dimension in emotion analysis, not limited to text. On this dataset, a study by Yinan Zhou[25], using XL-NET and Lemma preprocessing with early stopping, achieved an F1 score of 59.08%, yet this study also only considered the lyric dimension and not the audio aspect.

In the area of word embedding, studies have shown different embeddings have varied performances for different tasks. Word2Vec and GloVe are often used by researchers in emotion analysis studies[26][27][28] and have shown good performance.

Regarding audio features in music, Spotify provides a rich set of audio features to help analyze and understand music tracks. A study found that using audio features from Spotify and evaluating the performance of logistic regression in predicting song emotions achieved an F1 score of 86%[29]. Additionally, another study[30] used vector distance calculation combined with Spotify's Valance and Energy values, referencing Russell's emotional model, to more accurately define music's emotional classification. However, these studies only considered the emotional dimension of audio features.

Therefore, this research project aims to combine the analysis of lyrics text with the audio features provided by Spotify to further analyze and understand the influence of these two modalities on MER. In addition, it uses advanced machine learning techniques to fuse these two modalities and improve the accuracy and generalization ability of the model. In summary, the proposed method embodies a comprehensive effort to integrate and optimize multiple data sources for more effective music emotion prediction.

# CHAPTER 3

# Methodology

## 3.1 Dataset

### 3.1.1 Dataset 1(MoodyLyrics)

MoodyLyrics[22] is a public dataset built specifically for music sentiment analysis and contains the lyrics text of 2595 songs. The main feature of this dataset is that according to Russell's emotion model (Figure 3.1), song lyrics are classified based on their content words and their valence and arousal norms in the emotion lexicon, and labeled in four different emotion quadrants. This approach is unique in that it identifies the emotional content of a song based solely on textual analysis, regardless of the melody, rhythm, or sound characteristics of the music. As such, MoodyLyrics offers a unique perspective that focuses on understanding how lyrics themselves convey emotions.

### 3.1.2 Dataset 2(MoodyLyrics4Q)

MoodyLyrics4Q[24] is another publicly available dataset designed for music sentiment research and contains the lyrics of 2000 songs. This dataset is characterized by the fact that it utilizes Last.fm's user tags to classify songs for sentiment, where each song is labeled with a sentiment category in Russell's model (Figure 3.1). Compared to MoodyLyrics, MoodyLyrics4Q relies more on user-generated tags, which reflect the emotional perception and reactions of a broad audience. Therefore, this dataset provides an emotion recognition method that is closer to the

actual experience of listeners and enables researchers to understand music emotions from different perspectives.

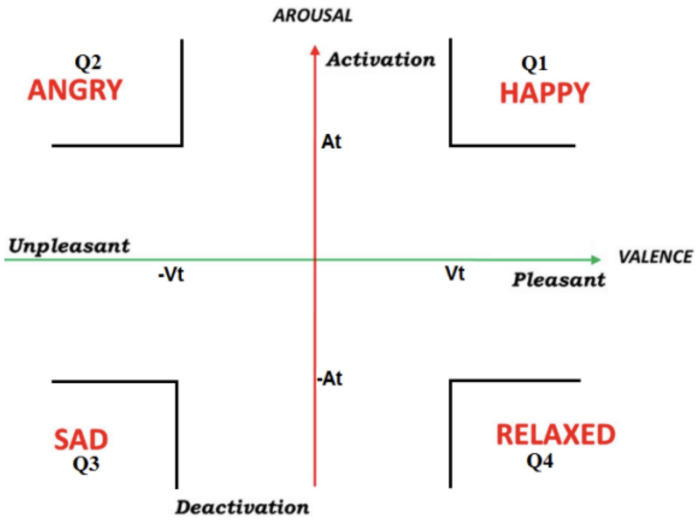


Figure 3.1: Russel Model

### 3.1.3 Data Collection

#### Lyrics Collection

Due to copyright restrictions, the original datasets did not include lyrics, necessitating a separate process for their acquisition. The initial approach involved using the Genius API, which is a service providing access to a vast collection of song lyrics.

The Genius API, accessed through the lyricsgenius package in Python, allows for the retrieval of song lyrics based on song titles and artist names. This method seemed promising as it provides a straightforward way to programmatically access accurate and legally available lyrics.However, it quickly became apparent that this method was prone to inaccuracies, especially in cases where song titles or artist names had multiple spellings.

To enhance the precision of lyric matching, improvements were made to a custom web scraping approach. This involved using the Google API to perform targeted searches on the Genius website and then parsing the HTML files of the returned results. The focus was on locating and returning the 'div' element in the HTML that contained the correct lyrics, ensuring a match with the artists and song titles from the dataset. The approach proved to be more effective in accurately locating and verifying song and artist information, significantly reducing the risk of retrieving incorrect lyrics. Additionally, it minimized the presence of noise commonly found in lyrics from the lyricsgenius API, such as "[Verse1]" tags and "\d+ Contributors," thereby reducing the need

for subsequent normalization and regularization steps. Although this custom approach was more labor-intensive, it provided a higher degree of accuracy in the data collection phase.

#### Audio Feature Collection

With the lyrics data in place, the next focus was on the acquisition of audio features for each track. This was accomplished through the Spotify API, a comprehensive and robust interface provided by the Spotify music streaming service.

The Spotify API offers a wide range of information about songs, albums, and artists. Of particular interest to this project were the audio features that Spotify provides, which include measurable aspects of a track like tempo, key, energy, danceability, loudness, and valence. These features are quantified based on Spotify's audio analysis algorithms.

To retrieve these audio features, we performed queries to the Spotify API using the song titles and artist names. This process involved sending a request to the API with the relevant track details and parsing the response to extract the desired audio features.

### 3.1.4 Data Cleaning and Standardization

After collecting the dataset, the next crucial step was data cleaning and standardization, which was essential for ensuring the quality and reliability of the data for subsequent analysis.The primary tool employed for this process was custom regular expressions. Regular expressions are extremely powerful for text processing and pattern matching, enabling the identification and removal of specific text patterns from the data efficiently.

**Removal of Non-Essential Information:** This included the elimination of elements like "[Verse1]" tags, which are often found in lyrics but do not contribute to the overall analysis. Such tags are more about the structure of the song rather than its content and can introduce noise into the data.

**Exclusion of Non-English Lyrics:** Since the analysis was focused on English songs, lyrics in other languages were removed to maintain consistency and relevance in the dataset.

**Correction of Erroneous Audio Features:** Any incorrect or outlier values in audio features provided by the Spotify API were identified and corrected or removed. This step was crucial to maintain the integrity and reliability of the dataset for analysis.

The implementation of these cleaning and standardization steps was pivotal in enhancing the dataset's overall quality, thereby laying a solid foundation for the analytical models and interpretations that followed.

### 3.1.5 Data balance

Upon completion of the collection and standardization processes, the dataset comprised a total of 2123 songs. However, it was observed that the dataset was imbalanced, with Dataset 1 showing a skew towards certain categories. To address this issue:

**Downsampling for Balance:** In Dataset 1, downsampling techniques were applied to achieve balance. Specifically, 90 'Happy' songs were randomly removed. A specific random state was used to ensure that the process is reproducible, thereby maintaining the integrity of the dataset for further analysis.

**Random Shuffling for Unbiased Training:** To prevent the model from learning any potential order in the data, random shuffling of the entire dataset was implemented prior to the training process. This was done using a specific random state to ensure consistency in the shuffling process, making it an unbiased step towards more reliable model training and evaluation.

#### Final Dataset Composition

After completing the data cleaning and standardization processes, the datasets were finalized with the following composition:

**Dataset 1:** The Final Dataset 1 contains 2033 records distributed across different moods:

- Happy: 554 records (27.2%)
- Relaxed: 532 records (26.2%)
- Angry: 501 records (24.6%)
- Sad: 448 records (22.2%)


Figure 3.2: Dataset1 Word Cloud

**Dataset 2:** The Final Dataset 2 contains 1576 records, relatively balanced across moods:

- Happy: 394 records (25.2%)
- Relaxed: 396 records (25.4%)
- Angry: 396 records (25.4%)
- Sad: 375 records (24%)


Figure 3.3: Dataset2 Word Cloud

**Dataset Structure**

The dataset is structured with multiple fields, each capturing a specific aspect of the song. Below is a horizontal table displaying the fields:

| Artist | Title | Mood | Lyrics | Danceability | Audio Features... |
|---|---|---|---|---|---|
| Usher | There Goes My Baby | Relaxed | There goes my baby... | 0.626 | 0.52, ... |

Table 3.1: Dataset Structure

## 3.2 Reproducing the paper

After preparing the data, I conducted a comprehensive literature review, which led me to a significant study by Jiddy Abdillah et al.[23], titled "Emotion Classification of Song Lyrics using Bidirectional LSTM Method with GloVe Word Representation Weighting". A key part of my research was the replication of this study, which utilized Dataset 1.The main goal of this replication was to assess the reliability and validity of the paper's findings and to consider its potential as a benchmark for my own research. This replication was not solely about confirming the reproducibility of the results; it also provided deeper insights into the methodologies and logical

frameworks of the original study. Additionally, this process allowed me to identify potential areas for improvement and issues within the context of using Dataset 1.

In line with my research, which also utilizes similar algorithms, and considering the replicated study's application of these methods, I decided to delve deeper into the explanation of these algorithms and their embedding techniques, particularly focusing on their role in text analysis. This exploration is crucial, as it helps establish a solid foundation for understanding how these methods contribute to sentiment analysis and emotion classification in song lyrics.

### 3.2.1 GloVe6B-100d

In their study, the authors employed the GloVe 6B-100d model to generate word embeddings. This model, part of the GloVe (Global Vectors for Word Representation)[31] project developed at Stanford University, provides 100-dimensional vector representations for words, trained on a 6 billion word corpus. These embeddings excel at capturing intricate semantic and syntactic patterns in text, exemplified by equations like "king - man + woman = queen (Figure 3.4)". The training objective of GloVe is to align the dot product of word vectors with the logarithm of their co-occurrence probabilities, effectively linking the logarithm of probability ratios to vector differences in the word vector space.

GloVe developed by Pennington et al. at Stanford. It combines matrix factorization and word context to generate word embeddings. GloVe constructs a co-occurrence matrix that counts how frequently words appear together in a given corpus, and then uses matrix factorization techniques to reduce the dimensionality of this matrix. This method effectively captures global statistical information about word relationships but may not be as sensitive to the local context of word usage as Word2Vec.

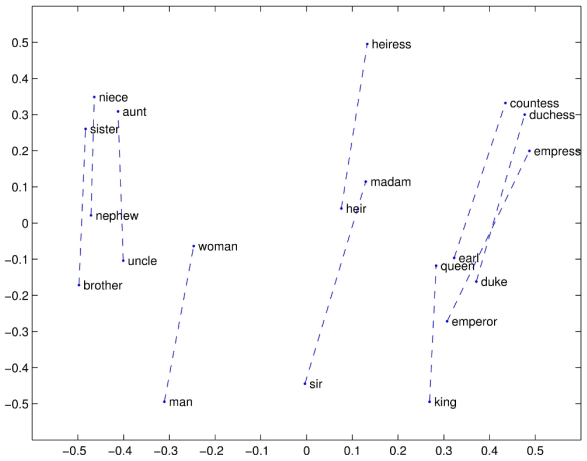


Figure 3.4: Glove Linear Substructures

### 3.2.2 Naive Bayes

Naive Bayes, stemming from Bayes' Theorem developed by Thomas Bayes in the 18th century, gained prominence as a fundamental classification technique in the 1960s when it was introduced under a different name in the field of text information retrieval[32]. Renowned for its efficiency with large datasets, such as those in sentiment analysis of song lyrics, NB is effective for categorizing text into different emotional tones using categorical data. It calculates the product of the prior probability of an emotional tone $P(c)$ and the likelihood of lyrical features given that tone $P(f_i|c)$. The most likely emotional tone $\hat{c}$ is identified by maximizing this product, demonstrating the method's simplicity and effectiveness in classifying text.

$$\hat{c} = \arg\max_{c \in C} \left\{ P(c) \prod_{i=1}^{n} P(f_i|c) \right\}$$

.

In the lyric sentiment analysis of the original paper, a special variant of Naive Bayes, namely Multinomial Naive Bayes, is often used due to its suitability for classifying discrete features such as word frequencies. The formula for Multinomial Naive Bayes applied to lyric sentiment analysis is:

$$P(C_k|\mathbf{x}) = \frac{P(C_k) \prod_{i=1}^{n} P(x_i|C_k)^{a_i}}{P(\mathbf{x})}$$

Here, $\mathbf{x}$ represents the frequency vector of words in lyrics, and $a_i$ indicates the frequency of word $i$ in the lyrics to be classified. The numerator $P(C_k) \prod_{i=1}^{n} P(x_i|C_k)^{a_i}$ calculates the likelihood of the lyrics belonging to a particular emotional class $C_k$, while the denominator $P(\mathbf{x})$ acts as a normalizing factor. In the lyric sentiment analysis, a smoothing parameter $\alpha$ was set to 0.05 to adjust for words not present in the training set and to prevent zero probabilities in subsequent computations.

### 3.2.3 K-Nearest Neighbors

K-Nearest Neighbors, initially introduced as a non-parametric method in the early 1950s and further developed in 1967 by Cover and Hart[33], is a supervised learning algorithm widely used in sentiment analysis of song lyrics. It is applicable to both classification and regression tasks, offering simplicity and intuitiveness in understanding and implementation. KNN's non-parametric nature enables it to handle various types of data distributions effectively, making it particularly suitable for diverse scenarios in natural language processing.

In the context of sentiment analysis, KNN determines the emotional tone of a song lyric by considering its nearest neighbors in the feature space. Each lyric is represented as a feature vector using GloVe embeddings to capture semantic information from the lyrics. KNN predicts the sentiment label based on the majority class or average value of the nearest neighbors in the training set. This proximity-based approach allows KNN to handle outliers and noise in the data, making it robust for sentiment analysis tasks.

Let $\mathbf{x}_q$ represent the query sample, $\mathbf{x}_i$ be a sample in the training set, and $d(\mathbf{x}_q, \mathbf{x}_i)$ denote the Euclidean distance between $\mathbf{x}_q$ and $\mathbf{x}_i$ in the feature space. The predicted sentiment label $\hat{y}_q$ for the query instance $\mathbf{x}_q$ is computed using a distance metric such as Euclidean distance (p=2).

$$\hat{y}_q = \text{argmax}_y \sum_{i=1}^{K} 1(y_i = y)$$

This formula calculates the predicted sentiment label $\hat{y}_q$ for $\mathbf{x}_q$ based on the sentiment labels of its $K$ nearest neighbors. In sentiment analysis of song lyrics, $\mathbf{x}_i$ represents the feature vector of the $i$th lyric instance, and $y_i$ is its sentiment label. In the replicated study, the parameter $K$ was set to 29, indicating that the algorithm considers the 29 nearest neighbors for sentiment classification using the Euclidean (p=2) distance metric, with lyric features represented through 100-dimensional GloVe embeddings.

### 3.2.4 Support Vector Machines

Support Vector Machines, developed by Boser, Guyon, and Vapnik in 1992[34], are a powerful and versatile supervised learning algorithm primarily used for classification and regression tasks. Known for their effectiveness in high-dimensional spaces, SVMs are particularly useful in situations where the number of dimensions exceeds the number of samples, such as in text classification using word embeddings like GloVe. In the sentiment analysis of song lyrics, as replicated in this study, SVMs are employed to classify lyrics into emotional categories. Each lyric is represented by a feature vector derived from GloVe 100d embeddings, capturing the semantic essence of the words. SVM aims to find a hyperplane in this 100d space that best separates different sentiment classes (Figure 3.5).

The decision function for a linear SVM is given by:

$$f(x) = \text{sign}(w \cdot x + b)$$

Here, $w$ is the weight vector determined by the SVM, $x$ denotes the input feature vector repre-

sented by GloVe100d embeddings, $b$ is the bias term, and the class label of $x$ is determined by the sign of the decision function $f(x) = \text{sign}(w \cdot x + b)$.

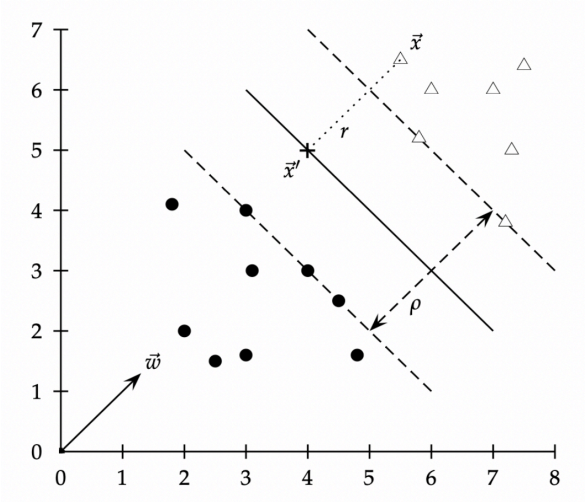


Figure 3.5: Visualization of the SVM Hyperplane

In the replicated study, the SVM algorithm was specifically configured with a linear kernel, reflecting the effectiveness of linear decision boundaries in high-dimensional spaces like those formed by GloVe embeddings. The penalty parameter $C$ was set to its default value of 1, as per the replicated paper, balancing the need for model complexity with the goal of regularizing the model. This setting helps in managing the trade-off between the model's ability to fit the data while keeping the model's complexity in check, ultimately aiding in effective generalization.

### 3.2.5 Convolutional Neural Networks

Convolutional Neural Networks, primarily developed for applications in computer vision, have also been effectively adapted for NLP. A seminal work in this adaptation is Yoon Kim's 2014 paper on "Convolutional Neural Networks for Sentence Classification,"[35] which demonstrated the efficacy of CNNs in text analysis tasks, including sentiment analysis of song lyrics.

CNNs apply convolutional layers to text data, treated as sequential like pixels in images. Each layer uses a set of filters or kernels to scan through word embeddings — numerical representations capturing the semantics of words. The filters, characterized by their kernel size, learn to recognize specific patterns in text. Smaller kernels capture local features (like n-grams), while larger ones understand broader contextual information. This enables CNNs to capture various levels of linguistic features, from simple word groupings to complex sentence structures, making them well-suited for diverse NLP applications (Figure 3.6).

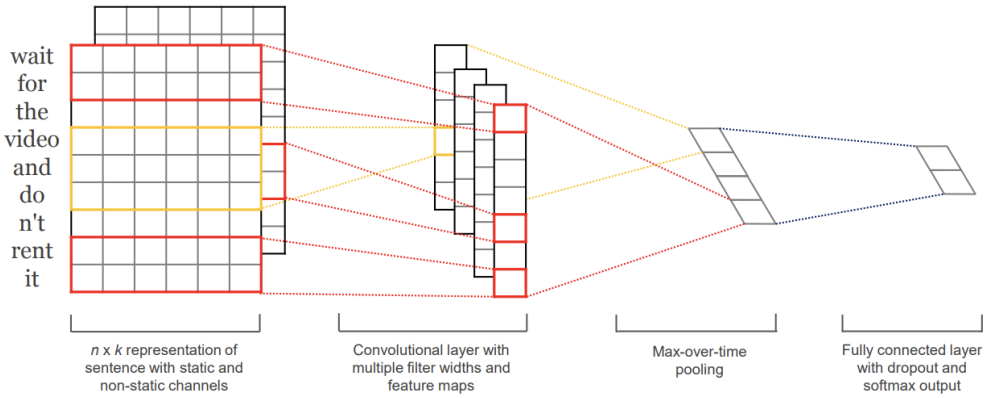


Figure 3.6: Illustration of Text-CNN architecture as proposed by Yoon Kim (2014).

In the replicated study for sentiment analysis of song lyrics, the CNN model was specifically designed with three one-dimensional convolutional layers, each equipped with 128 filters of kernel size 5 and the ReLU activation function. This architecture effectively processes the complex textual structures in song lyrics, with a maximum sequence length set to 1000. The model also includes three max-pooling layers with a pool size of 5, which help condense the feature maps and highlight salient features for classification. The final output layer utilizes the Softmax activation function to classify the lyrics into various emotional states. For the training of the CNN model, the ADAM optimizer with its default learning rate of 0.001 was used. The Training was conducted over 20 epochs with a batch size of 64, ensuring thorough learning while balancing computational efficiency.

### 3.2.6 Long Short-Term Memory and Bidirectional LSTM

Long Short-Term Memory networks, introduced by Sepp Hochreiter and Jürgen Schmidhuber in 1997[36], are a specialized form of RNN designed to capture long-term dependencies in sequential data. The distinctive feature of LSTM is its use of gating mechanisms that control the flow of information. These gates include:

- **Input Gate**: Determines how much of the new information to incorporate into the cell state.
- **Forget Gate**: Decides the amount of the previous cell state to retain.
- **Output Gate**: Controls the extent to which the value in the cell influences the output.

These gates allow LSTMs to selectively remember or forget patterns over time, which is

crucial for tasks that require understanding data over long time intervals, such as in language processing (Figure 3.7).

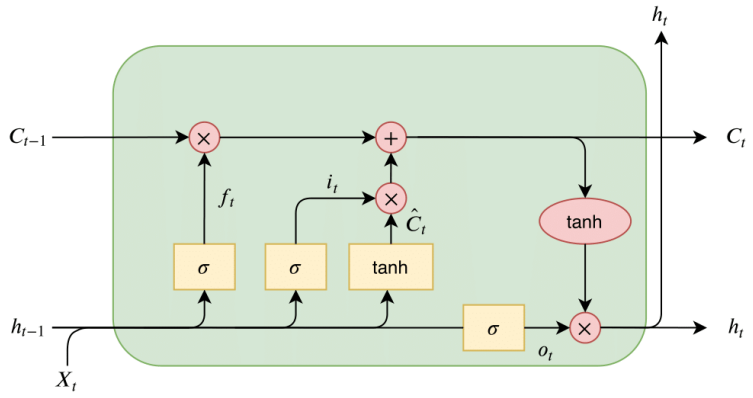


Figure 3.7: LSTM Structure

Expanding on the LSTM concept, the Bidirectional LSTM (Bi-LSTM) processes data in both forward and backward directions, allowing the model to have both past and future context. This feature is particularly useful in text analysis, where the context surrounding each word can significantly influence its meaning. The Bi-LSTM can be described by the following formulas, where $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ represent the forward and backward hidden states, respectively:

$$\overrightarrow{h_t} = H(W^f_{xh}x_t + W^f_{hh}\overrightarrow{h_{t-1}} + b^f_h)$$

$$\overleftarrow{h_t} = H(W^b_{xh}x_t + W^b_{hh}\overleftarrow{h_{t-1}} + b^b_h)$$

Here, $H$ is an activation function like tanh or ReLU, $W$ and $b$ are the weights and biases for each gate, and $x_t$ is the input at time step $t$. The final output at each time step $t$ in BiLSTM is then the concatenation of $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$, providing a comprehensive view of the sequence from both directions.

In the replicated study of song lyrics, both LSTM and Bi-LSTM models were configured with identical parameters to achieve optimal performance. The models utilized 100-dimensional GloVe embeddings for text representation, and included layers with 100 hidden units, tailored to process sequence lengths of up to 1000.The Training was conducted with a batch size of 64, and the models employed the Softmax activation function in the output layer for effective sentiment categorization. The ADAM optimizer was chosen for its efficiency, with a learning rate set at 0.006, ensuring a balanced convergence speed. Additionally, the training was performed over 20 epochs, allowing the models sufficient time to learn and adapt to the complexities of the lyrical content.

### Process of replicating

In the process of replicating the original study, I divided the dataset in accordance with the guidelines of the original research, allocating 80% for training and 20% for testing. The text preprocessing included tokenization, lemmatization (Lemma), removal of stopwords (SR), conversion to lowercase (LC), and noise removal (NR), aimed at standardizing and clarifying the data. For the NB model, I chose Tf-idf embedding, as the NB classifier typically assumes features are independently and non-negatively distributed, as seen in multinomial models. GloVe embeddings, which can contain negative values, are not commonly used in probabilistic models like NB, as standardizing or normalizing these embeddings to fit NB could potentially lead to a loss of some original data information. Therefore, Tf-idf embedding was selected to transform the text data into a suitable format for NB. In contrast, for other models including CNN, LSTM, Bi-LSTM, KNN, and SVM, I employed GloVe embeddings as described in the original paper. This adherence to specific embedding techniques ensured a uniform basis for model comparison and analysis, strictly aligning with the methodological design of the replicated study.

The purpose of this section is to detail the steps and decisions followed during the replication process to ensure a faithful reproduction of the original study, and to provide the necessary background for the subsequent results and analysis sections. Should the results closely mirror those of the original study, it would validate the effectiveness of the research and its authority as a benchmark. If there are significant discrepancies, I will explore the potential reasons for these differences, which may include questioning the methods of the original study or identifying issues in my implementation process. The results of this replication will be presented in Chapter 4.

Building upon the current replicated study, three key objectives have been identified for potential enhancements in sentiment analysis of song lyrics. These objectives are:

1. **Word embedding**: To explore alternative text embedding methods that may offer a more nuanced understanding and representation of lyrics.

2. **Text Preprocessing**: To refine and potentially develop more effective preprocessing strategies that could lead to better feature extraction and model accuracy.

3. **Audio Feature**: To investigate how incorporating audio data can augment sentiment analysis, providing a more holistic view of the songs beyond just textual content.

Each objective is aimed at addressing specific limitations or areas of improvement identified in the replicated study, with the goal of advancing the robustness and depth of sentiment analysis approaches.

## 3.3 Word embedding

As part of the methodology for the first objective focused on embedding techniques, an initial analysis of the original data distribution was conducted. After applying the same preprocessing methods as the original study, including tokenization, Lemma, LC, SR and NR, it was observed that 97% of the lyrics in the dataset had a sequence length of 250 words or fewer post-processing. This insight led to the realization that the 1000 word sequence length used in the original paper might introduce excessive padding, potentially adding noise to the dataset. Consequently, for deep learning models, the sequence length was adjusted to 250 to better match the actual distribution of the lyrics in the dataset (Figure 3.8).

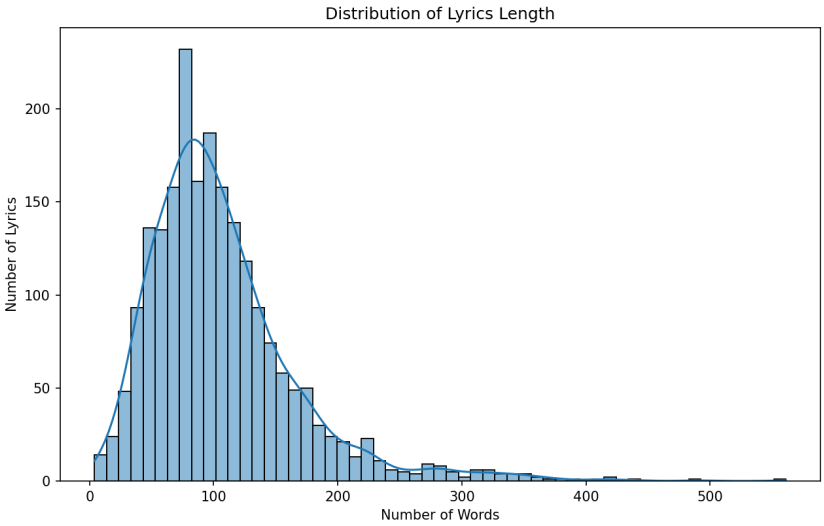


Figure 3.8: Distrilbution of lyrics

In addressing the unique features of song lyrics, particularly their repetitive and rhythmic nature, the research focused on selecting suitable embedding techniques. To this end, Bag of Words, Tfidf, and Word2Vec were chosen. These methods were anticipated to be more effective in capturing the essence of lyrical content than GloVe embeddings, which are generally oriented toward global statistical analysis and might not sufficiently represent localized and repetitive lyric patterns. This selection was informed by research such as "Rhythm and Myth in the Lyrical Genre" by G. Oripova[37] and "Modeling Discourse Segments in Lyrics Using Repeated Patterns," by K. Watanabe[38] which highlight the critical nature of these features in song lyrics. The detailed rationale behind the choice of these specific embeddings and their expected alignment with the characteristics of song lyrics will be explored in subsequent sections of the research.

### 3.3.1 BoW

The Bag of Words model[39], with its straightforward approach to text representation, offers specific advantages when applied to the analysis of song lyrics. Given that song lyrics often contain repetitive patterns and keywords that are crucial for understanding the overall sentiment and theme, BoW can be particularly effective in capturing these essential elements.

In the BoW model, each unique word in a set of lyrics is treated as a feature, and the document is represented as a vector indicating the frequency of each word. This representation aligns well with the structure of the lyrics, as it emphasizes the presence and recurrence of significant words. For instance, in a song expressing joy or sorrow, words associated with these emotions will likely appear multiple times, and their frequency can be a strong indicator of the overall sentiment (Figure 3.9).

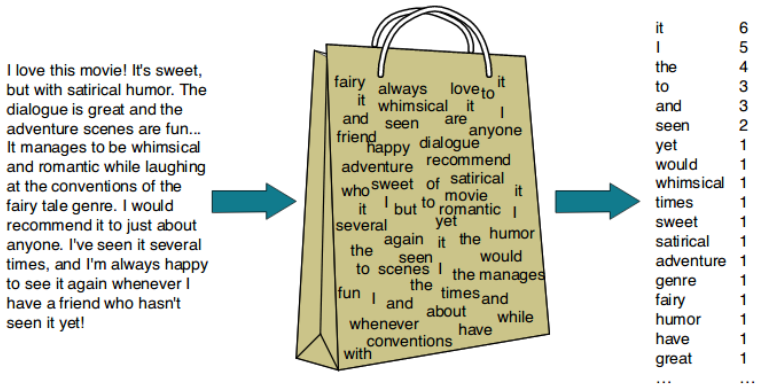


Figure 3.9: BoW

Mathematically, if we consider a song's lyrics as a document $D$ and our vocabulary as $V$ consisting of words $w_1, w_2, \ldots, w_n$, the BoW representation of $D$ is a vector $[x_1, x_2, \ldots, x_n]$, where $x_i$ is the frequency of word $w_i$ in $D$. For example, if the vocabulary is ["love", "heart", "joy", "sad"], and the lyrics include "love" and "heart" multiple times but not "joy" or "sad", the BoW vector might look like $[2, 3, 0, 0]$, indicating the frequencies of these words in the song. This is computed as:

$$x_i = \text{freq}(w_i, D)$$

where $\text{freq}(w_i, D)$ is the number of times $w_i$ appears in $D$.

While the BoW model offers simplicity and computational efficiency in analyzing song lyrics, it's important to recognize its limitations. A primary shortcoming of BoW is that it treats all words

with equal importance, without considering their frequency across different documents. This can result in common words being given the same weight as more unique and potentially more meaningful words in the context of lyrics. BoW's focus on mere word presence fails to capture the nuanced significance that some words may hold over others in conveying the emotional tone of a song.

### 3.3.2 Tf-idf

Term Frequency-Inverse Document Frequency[40] is an advanced text representation method that improves upon the basic BoW approach. Tf-idf combines the raw frequency of a word (Term Frequency, TF) with its inverse frequency in the entire document corpus (Inverse Document Frequency, IDF), thereby balancing word frequency with its uniqueness.
Mathematically, the TF of a word is simply the number of times it appears in a document, normalized by the total number of words in that document. The formula for TF is given as:

$$\text{TF}(w, D) = \frac{\text{Number of times word } w \text{ appears in document } D}{\text{Total number of words in document } D}$$

IDF, on the other hand, is calculated as the logarithm of the number of documents divided by the number of documents containing the word. The formula for IDF is:

$$\text{IDF}(w) = \log\left(\frac{\text{Total number of documents}}{\text{Number of documents with word } w}\right)$$

The Tf-idf score of a word in a document is then the product of its TF and IDF scores:

$$\text{Tf-idf}(w, D) = \text{TF}(w, D) \times \text{IDF}(w)$$

For instance,consider a set of song lyrics with a vocabulary consisting of the words ["love", "heart", "joy", "sad"]. If in a particular song the frequencies of these words are [2, 1, 3, 0] respectively, and their corresponding IDF values are [1.2, 1.5, 1.0, 1.7], the Tf-idf vector would be computed as [$2 \times 1.2$, $1 \times 1.5$, $3 \times 1.0$, $0 \times 1.7$], resulting in a vector representation of [2.4, 1.5, 3.0, 0]. This vector effectively captures the importance and frequency of each word in the context of the song lyrics, providing a more nuanced understanding of the lyrics' emotional content.

This scoring method is particularly effective in the context of song lyrics analysis. Lyrics often contain repetitive phrases and common words that, while important, may not contribute significantly to the overall sentiment or theme of the song. Tf-idf helps to attenuate the weight of these common words (is, or, and) while highlighting unique or rare words that could be more

indicative of the song's emotional and thematic content. By focusing on these significant words, Tfidf provides a more nuanced understanding of the lyrics, capturing not just the frequency of word usage but also its relative importance in the broader context of the song and the corpus.

### 3.3.3 Word2Vec

Word2Vec is a popular word embedding technique in natural language processing. Word2Vec, developed at Google by Mikolov et al[41], uses a neural network model to embed words from a text corpus into a high-dimensional space. It essentially learns word representations either by predicting the context of the word (in Skip-gram) or according to the context (in CBOW)(Figure 3.10). This approach captures the nuances of word usage in a particular context, making it particularly effective for understanding the meaning of words used in real-world text. When applied to song lyrics, which often feature repetitive and rhythmic patterns, Word2Vec's context-focused learning offers the potential for deeper insights. It can identify the subtle ways in which words are used within the unique structure of lyrics, potentially providing a more thorough understanding of their thematic and emotional layers.

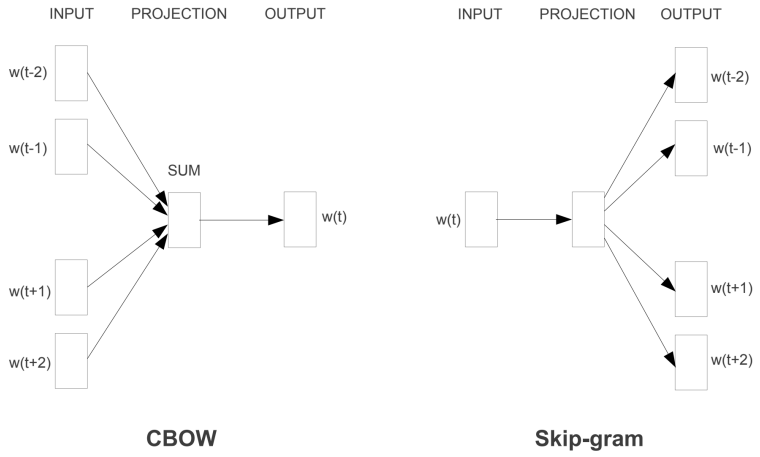


Figure 3.10: CBOW and Skip-gram

For instance, in emotion analysis of song lyrics using Word2Vec, a word like "melody" might be represented by a 300-dimensional vector such as $[0.08, -0.21, 0.09, ..., -0.05]$. Each dimension in this vector represents a unique feature that captures some aspect of how "melody" is related to various emotions based on the training data. For example, one dimension might capture its association with Happy contexts, another with Sad sentiment. This comprehensive representation allows for a deeper understanding of the emotional connotations and nuances associated with words in lyrics.

While GloVe excels in capturing broad word relationships, Word2Vec's emphasis on context may offer a unique advantage, especially for analyzing lyrics. Lyrics, with their inherent structure, often rely on repetition and rhythm to convey emotions and narratives. The context-aware nature of Word2Vec potentially allows for a deeper understanding of the emotional and thematic expressions in lyrics. To capitalize on this, I utilized a pre-trained 300-dimensional Word2Vec model for embedding. This model, developed by a team at Google and trained on a massive corpus of data, encompasses a wide range of lexical semantics and syntactic information. The depth and breadth of its training enable it to capture nuanced meanings and associations between words, which is particularly beneficial for the complex and often metaphorical language found in song lyrics. The 300-dimensional vectors provide a detailed and rich representation of words, thus offering a more refined understanding of the lyrical content.

#### Model

In the word embedding experiment, I applied a uniform preprocessing method Lemma+SR+NR+LC across all models. This consistency was crucial for accurately assessing the impact of different embedding techniques. For traditional models like NB, SVM, and KNN, I employed BoW and Tf-idf embeddings. In contrast, for neural network models, including CNN, LSTM, and Bi-LSTM, I used Word2Vec300d embeddings for deep semantic understanding. To optimize these models, I utilized Grid Search for the traditional models and learning curve analysis for the neural networks. Specifically for the CNN model, early stopping was used to find the optimal number of epochs, and an exponential decay learning rate was applied to enhance training stability and prevent overfitting. This rate decreases over time according to the formula: Learning Rate $=$ Initial Learning Rate $\times$ $e^{(-\text{decay rate}\times\text{epoch})}$, ensuring that adjustments to the model become more refined as training progresses. The goal was to evaluate how each embedding technique influenced the models' performance in processing lyrical content. Chapter 4 will present the detailed results and performance analysis of these embedding techniques across various models, demonstrating their effectiveness in sentiment analysis of song lyrics.

## 3.4 Text Preprocessing

Following the outcomes of the embedding experiments, I selected four models that demonstrated the best performance for further investigation: Tf-idf-SVM, BoW-NB, Word2Vec-CNN, and Word2Vec-Bi-LSTM. A fundamental and common step in the preprocessing stage for all these models was tokenization, a process where text is broken down into smaller units, typically words,

which makes it possible for machines to understand and analyze the text effectively.

Preprocessing, especially tokenization, is essential in natural language processing as it directly impacts how text data is interpreted and analyzed. For example, consider the lyric line 'Echoes of heartbeats, rhythms in the dark.' When tokenized, it becomes: ['Echoes', 'of', 'heartbeats', 'rhythms', 'in', 'the', 'dark']. For song lyrics, which often contain rich and nuanced expressions, accurate tokenization ensures that each word's sentiment and meaning are appropriately captured and assessed. This step, along with other preprocessing methods like Stemming, Lemmatization, Lowercasing (LC), Noise Removal (NR), and Stopword Removal (SR), helps refine the input data, allowing the models to focus on the most relevant aspects of the lyrics. These preprocessing techniques collectively contribute to the accuracy and efficiency of the sentiment analysis, providing a clearer insight into the emotional tone and nuances present in song lyrics.

### 3.4.1 Stemming

Stemming reduces words to their base or root form, which can be particularly useful in analyzing song lyrics where different forms of a word convey the same sentiment or theme. For example, stemming transforms variations like "loving", "loved", and "loves" to the root word "love", simplifying the analysis of recurring themes in lyrics.

### 3.4.2 Lemmatization

Lemmatization considers the word's part of speech and context to reduce it to its dictionary form. This is essential in understanding the nuanced meanings in song lyrics, where words like "better" might carry different emotional weights, and are more accurately represented when lemmatized to "good".

### 3.4.3 Lowercasing

Lowercasing is crucial in processing lyrics as it addresses the issue of inconsistent capitalization. It ensures words like "Moon" and "moon" in the lyrics are treated uniformly, facilitating consistent sentiment analysis.

### 3.4.4 Noise Removal

Noise removal is key in cleaning song lyrics, especially when sourced from various platforms. It involves removing non-linguistic elements, like special characters or formatting symbols, which might interfere with the textual analysis.

### 3.4.5 Stopword Removal

In song lyrics, certain words, although frequently used, don't contribute much to the overall sentiment or theme. Removing these stopwords, such as "the", "is", and "at", allows for a more focused analysis on meaningful words, enhancing the understanding of the lyrical content.

To conduct a comprehensive evaluation of preprocessing methods for sentiment analysis of song lyrics, I established a systematic testing loop encompassing four models: Tfidf-SVM, BoW-NB, Word2Vec-CNN, and Word2Vec-BiLSTM. This loop was designed to focus on examining various preprocessing combinations, involving techniques such as Stemm, Lemma, LC, NR, and SR. The primary objective was to assess the impact of these preprocessing methods on the models' ability to accurately analyze and interpret the subtle complexities of lyrical content, identifying which preprocessing approach most effectively captures the sentiment of the lyrics. For deep learning models like CNN and Bi-LSTM, I employed cross-validation and multiple iterations of testing to mitigate the effects of random weight initialization, ensuring the reliability and robustness of performance outcomes under different preprocessing scenarios.
The following Tables 3.2 and 3.3 illustrate the preprocessing combinations tested in this research.

| **Preprocessing Methods (Lemma)** | **Preprocessing Methods (Stem)** |
|---|---|
| Lemma + LC + NR + SR | Stem + LC + NR + SR |
| Lemma + NR + SR | Stem + NR + SR |
| Lemma + NR + LC | Stem + NR + LC |
| Lemma + SR + LC | Stem + SR + LC |
| Lemma + NR | Stem + NR |
| Lemma + SR | Stem + SR |
| Lemma + LC | Stem + LC |
| Lemma | Stem |

Table 3.2: Combinations of Preprocessing Methods(Lemma,Stem)

| **Preprocessing Methods (SR)** | **Preprocessing Methods (NR)** | **Preprocessing Methods (LC)** |
|---|---|---|
| SR + LC + NR | NR + LC + SR | LC + NR + SR |
| SR + NR | NR + SR | LC + SR |
| SR + LC | NR + LC | LC + NR |
| SR | NR | LC |

Table 3.3: Combinations of Preprocessing Methods(SR,NR,LC)

The results of the preprocessing steps will be presented in Chapter 4.

## 3.5 Audio feature

In exploring sentiment analysis of song lyrics, I realized that relying solely on textual information could be limiting. The emotional content of a song is not only conveyed through its lyrics but also significantly through its musical elements such as melody, rhythm, and tonality. Therefore, to gain a more comprehensive understanding of emotional expressions in songs, I decided to integrate audio features into the analysis.

To achieve this, I utilized a suite of audio features provided by Spotify. These audio features from Spotify represent quantitative metrics that disclose various musical aspects of a track. 'Danceability,' for instance, gauges a track's suitability for dancing by examining its tempo stability, beat strength, and rhythmic regularity. 'Energy' measures the intensity and activity of a track through its dynamics, loudness, and timbre. Additional features, such as 'Key' and 'Loudness,' provide quantitative measures of a track's tonal center and its overall volume level.

The following Table 3.4 lists the primary audio features provided by Spotify along with their descriptions:

Table 3.4: Spotify Audio Features

| **Spotify Audio Feature** | **Description and Range** |
|---|---|
| Danceability | Describes a track's suitability for dancing, based on tempo stability and beat strength. Range: 0.0 to 1.0. |
| Energy | Measures a song's intensity and activity, considering dynamics, loudness, and timbre. Range: 0.0 to 1.0. |
| Key | Indicates the key of the track, with values representing different musical keys. Range: 0 to 11. |
| Loudness | Represents the overall loudness of a track in decibels (dB). Range: Typically between -60 and 0 dB. |
| Speechiness | Detects the presence of spoken words in a track, with higher values indicating more speech. Range: 0.0 to 1.0. |
| Acousticness | Measures the acoustic quality of a track, with higher values indicating more acoustic sounds. Range: 0.0 to 1.0. |
| Instrumentalness | Predicts whether a track contains no vocals, with higher values indicating less likelihood of vocals. Range: 0.0 to 1.0. |

Continued on next page

Table 3.4 continued from previous page

| Spotify Audio Feature | Description and Range |
|---|---|
| Liveness | Detects the presence of an audience in the recording, with higher values indicating a live performance. Range: 0.0 to 1.0. |
| Valence | Indicates the musical positiveness conveyed by a track, with higher values suggesting happier and more positive music. Range: 0.0 to 1.0. |
| Tempo | Measures the overall tempo of a track in beats per minute (BPM). Range: BPM values (118.211). |
| Duration_ms | Provides the length of the track in milliseconds. Range: Variable, in milliseconds (237040). |
| Time Signature | Specifies the type of meter a track is in, indicating the rhythmic pattern. Range: Common values include 3, 4, etc. |
| Mode | Indicates the modality of the track, where major is represented by 1 and minor is 0. Range: 0 (minor) or 1 (major). |

By integrating these audio features, my analysis goes beyond the textual dimension to offer a richer interpretation of emotions from a musical perspective, thereby capturing and understanding the overall sentiment of songs more accurately.

### 3.5.1 Spotify Audio Data Analysis

#### Audio Data Preprocessing and Standardization

In the realm of audio feature analysis, preprocessing and standardization of data are vital steps. The heterogeneity and scale variance inherent in audio data necessitate these steps to ensure consistent and meaningful model inputs. Standardization, in particular, is essential in transforming diverse audio features into a format conducive for machine learning models to process effectively. Standardization was performed on continuous audio features using the formula:

$$z = \frac{(x - \mu)}{\sigma}$$

where $x$ is the original data value, $\mu$ is the mean, and $\sigma$ is the standard deviation. This approach normalizes the range of continuous features, such as 'Duration_ms' and 'Loudness',

thereby enabling each feature to contribute proportionately to the final prediction. It circumvents the risk of features with larger scales disproportionately influencing the model's learning process, promoting more accurate and stable predictions. Post-standardization, these key audio features retain their interpretability, a critical aspect in analyzing and understanding song moods.

For categorical features like 'Key' and 'Mode', one-hot encoding was implemented. This process transforms these categories into a binary format that machine learning models can interpret and utilize effectively. One-hot encoding expands the feature space to represent each category distinctly, ensuring that the categorical nuances of audio data are preserved and accurately represented in the analysis process.

Through these preprocessing and standardization steps, the audio data was rendered into a state ideal for subsequent analysis, facilitating a more precise and insightful exploration of the emotional undertones conveyed through the audio features of songs.

#### Audio Data Structure and Clustering Tendency Analysis

After standardizing Dataset1, a crucial step was to understand its clustering tendency, which is pivotal for guiding the analytical approach in sentiment analysis. Clustering tendency refers to the dataset's propensity to form distinct groups or patterns, an essential consideration for models focusing on pattern recognition and classification. To assess this tendency, particularly in the high-dimensional space of Dataset1, I utilized the Hopkins statistic[42]. This method effectively determines if the dataset demonstrates significant groupings beyond random distribution, which can be instrumental in choosing suitable modeling techniques and approaches for the analysis.

The process involved generating random points, rand_X, uniformly distributed across each dimension of the dataset. Distances $u$ from these points to their nearest neighbors within the dataset were computed, and contrasted with distances $w$ from randomly selected data points to their second nearest neighbors. The Hopkins statistic was calculated as:

$$H = \frac{\text{mean}(u)}{\text{mean}(u) + \text{mean}(w)}$$

where $u$ represents the distances from high-dimensional random points to their nearest neighbors, and $w$ the distances from actual data points to their second nearest neighbors.

Upon applying the Hopkins statistic to Dataset 1, the values consistently hovered around 0.75. This outcome suggested the presence of potential clusters within the dataset, indicating that the audio features might not be randomly distributed. Acknowledging this tendency, I proceeded to employ PCA and T-SNE for further exploration and visualization of these potential clusters

within the dataset.

### Audio High-Dimensional Data Visualization

Following the indication of clustering tendencies in Dataset 1, as revealed by the Hopkins statistic, I proceeded with high-dimensional data visualization. This step was critical in further exploring and understanding the underlying structures within the audio features. For this purpose, I employed two distinct techniques: Principal Component Analysis (PCA)[43] and t-Distributed Stochastic Neighbor Embedding (T-SNE)[44].

### Principal Component Analysis (PCA)

PCA is a linear dimensionality reduction technique that reduces the dimensionality of data by identifying principal components.It starts by calculating the covariance matrix of the data. The principal components are then extracted as the eigenvectors of this covariance matrix, ordered by their corresponding eigenvalues. Mathematically, this process can be represented as:

$$\text{Covariance Matrix: } \Sigma = \frac{1}{n-1}(X - \bar{X})^T(X - \bar{X})$$

$$\text{Principal Components: Eigenvectors of } \Sigma$$

where $X$ is the original data, and $\bar{X}$ is the mean-centered data.

### T-Distributed Stochastic Neighbor Embedding (T-SNE)

T-SNE is a non-linear dimensionality reduction technique particularly well-suited for embedding high-dimensional data into a space of two or three dimensions. It works by minimizing the Kullback-Leibler divergence between two distributions: one representing pairwise similarities of the input data points in the high-dimensional space, and the other representing pairwise similarities of the corresponding points in the lower-dimensional space. The cost function in T-SNE is given by:

$$C = \text{KL}(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

where $P$ represents the probability distribution in the high-dimensional space, and $Q$ represents the probability distribution in the low-dimensional space. The probabilities $p_{ij}$ and $q_{ij}$ correspond to the similarities between data points.

These visualization techniques, adept at depicting both linear and nonlinear aspects, enabled a comprehensive exploration of the complex, high-dimensional nature of audio data. This approach

provided valuable insights into underlying patterns and relationships, unveiling dimensions of data that might not be readily apparent through traditional analysis methods.

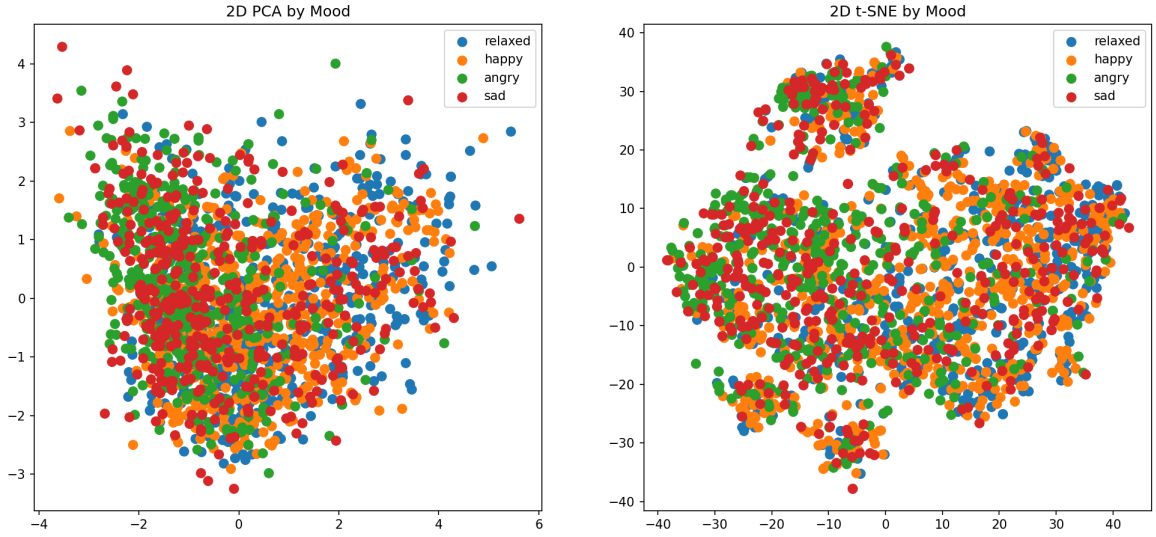


Figure 3.11: 2D PCA and t-SNE in Dataset 1

The visualization results from PCA and T-SNE in Dataset1 (Figure 3.11) indicated some degree of overlap within the entire dataset. However, a pattern emerged where most songs expressing "angry" tended to cluster towards the left side of the graph, while "happy" songs were predominantly positioned on the right side. This distribution might reflect significant differences in certain audio features between music of different emotional categories. For instance, "angry" songs might have distinct characteristics in certain tonal attributes that help group them on one side of the reduced dimensional space after PCA and T-SNE processing. Similarly, "happy" songs could have unique features affecting their placement on the opposite side of the space.

Combining these observations with the Hopkins statistic value, which was close to 0.75, indicates that while the dataset exhibits a clustering tendency in high-dimensional space, there is some overlap in the lower-dimensional space post-PCA and T-SNE processing. However, this overlap doesn't entirely negate the dataset's clustering characteristics in high dimensions. The overlaps might partly result from information loss during the dimensionality reduction process. These findings provide a foundation for further analysis, guiding me to use various tools and methods for a deeper understanding of the relationship between audio features and emotional categories.

### Audio Heatmap Analysis

To deepen the understanding of how various audio features might correlate with specific emotions in songs, Heatmap[45] analysis was utilized. This method offers a visual and intuitive means to observe the complex interactions among different audio features. By representing the Pearson correlation coefficient for pairs of audio features, Heatmaps facilitate the identification of significant relationships and patterns among the features, revealing their collective influence on the emotional content of a song. The cornerstone of this analysis, the Pearson correlation coefficient, quantifies the linear relationship between two variables and is essential for elucidating the connections between different audio features. This coefficient is defined as:

$$r = \frac{\sum(x - \bar{x})(y - \bar{y})}{\sqrt{\sum(x - \bar{x})^2 \sum(y - \bar{y})^2}}$$

where $x$ and $y$ represent the values of two different audio features, $\bar{x}$ and $\bar{y}$ are the mean values of these features.

The resulting coefficient $r$ varies between -1 and 1, where 1 indicates a strong positive correlation, -1 indicates a strong negative correlation, and 0 indicates no correlation. The heatmap visualizes these correlation values, with color intensity and patterns revealing the strength and direction of the relationships between features.

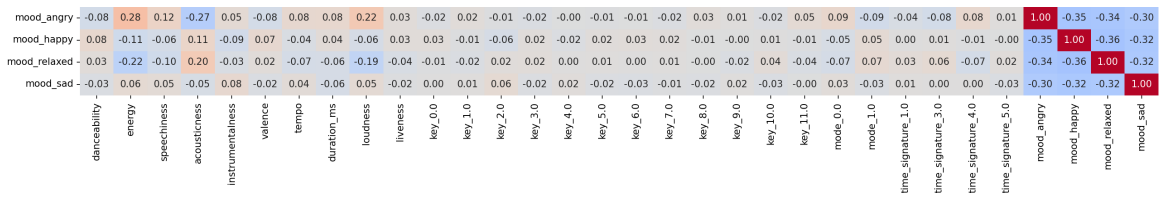


Figure 3.12: Heatmap in Dataset 1

This heatmap analysis in Dataset 1 yielded pivotal insights into the relationships among various audio features, especially in the context of their influence on specific moods in songs (Figure 3.12). For instance, a notable positive correlation was observed between 'Loudness' and 'Energy' in relation to the mood classified as 'Angry'. This correlation suggests that tracks with higher loudness and energy levels are more likely to be perceived as embodying anger or aggression. On the other hand, these same features exhibited a negative correlation with the 'Relaxed' mood, indicating that songs with lower loudness and energy levels tend to be associated with a more relaxed or calm emotional tone. These correlations highlight the nuanced ways in which different audio features can collectively shape the emotional character of a song, thereby enriching our understanding of the complex interplay between music and emotions.

### Audio Feature selection

Following the heatmap analysis, which highlighted certain audio features with weak correlations in mood prediction, I employed Random Forest[46] to ascertain the importance of each audio feature in predicting song moods. This method constructs multiple decision trees during training, outputting either the mode (for classification) or mean prediction (for regression) from each tree. The essential aspect of using RF was assessing feature importance to understand how various audio features influence a song's mood. This importance was gauged based on each feature's contribution to the predictive accuracy of each decision tree within the ensemble, with features that notably improved predictive power considered more important.

This process included analyzing the improvement in prediction accuracy when each feature was incorporated into the forest's trees. Features that split higher in the trees were assigned greater importance as they contributed more significantly to reducing uncertainty or impurity in the dataset. The RF algorithm provided quantifiable measures of each audio feature's importance, allowing me to rank and understand the relative significance of features like tempo, energy, and danceability in determining a song's emotional tone or mood. This meticulous approach enabled me to identify which specific audio characteristics most substantially impact the effective classification of song moods.

From the outcomes of both the RF feature importance (Figure 3.13) and heatmap analysis (Figure 3.12), it became evident that one-hot encoded features such as Key, Mode, and Time Signature showed very low correlation and importance. Additionally, these features significantly increased the dimensionality of the dataset, potentially leading to the curse of dimensionality. Therefore, I decided to exclude these three audio features from further analysis.

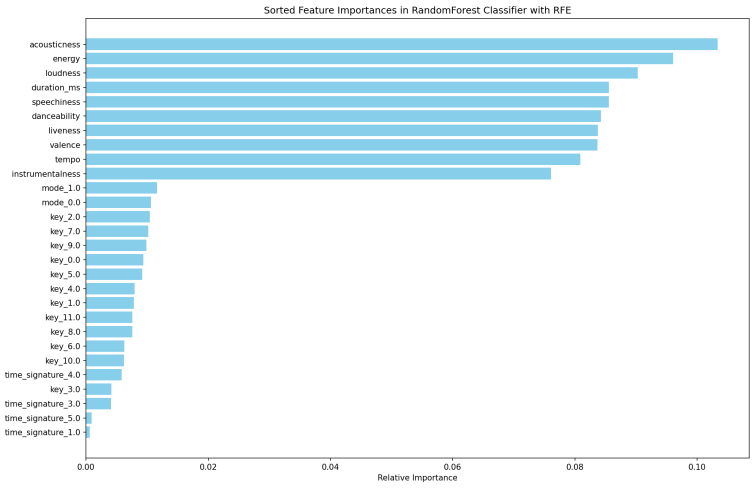


Figure 3.13: RF feature importance in Dataset 1

### 3.5.2 Integration of Spotify Audio Features

#### Initial Exploration with Audio Features only

After processing and analyzing the audio features, I employed SVM, dense neural network layers, and RF in my experiments to explore the potential of audio features alone in sentiment analysis. The choice of these methods was based on their unique strengths. This phase was crucial for establishing a baseline understanding of the independent contribution of audio elements to sentiment analysis. SVM was chosen for its effectiveness in high-dimensional spaces; dense layers in neural networks have the capability to recognize and process complex patterns through their interconnected structure; RF was included for its ensemble learning approach, which brings together multiple decision trees to enhance overall predictive accuracy and reliability while effectively reducing the risk of overfitting.

#### Integrated Approach with Textual and Audio Data

In building upon the textual analysis, I continued to integrate audio features into the existing models, aligning with my goal of enriching sentiment analysis through a blend of textual and audio insights. During this process, SVM was utilized for early feature fusion, combining audio and text data to take advantage of SVM's proficiency in high-dimensional spaces. This fusion method with SVM has the potential to correct misclassifications near decision boundaries by leveraging additional audio cues, thus enhancing overall accuracy. NB was excluded from this phase due to its limitation in handling negative values.

For the CNN and Bi-LSTM models, I integrated additional dense layers specifically for processing audio features before their fusion with the textual data. This bifurcated approach ensures that each data type is first understood independently, allowing the models to harness the distinctive characteristics of both lyrical content and audio signals. The dense layers assigned to handle audio data allow for a nuanced interpretation of musicality, which complements the lyrical analysis performed by the CNN and Bi-LSTM structures.In this way, the model is trained to detect complex patterns where lyric elements and audio elements are interwoven to enable a more sophisticated perception of the emotional expression in the song. The goal of this methodology is to exploit the synergistic potential of textual and audio information, thereby capturing a wider array of emotional indicators present in music.

### Model

In terms of individual model architectures, each model maintained its core structure from previous experiments. For SVM, early fusion was implemented using the same regularization techniques, ensuring a consistent approach to handling the combined data. The Bi-LSTM model's text processing component for lyrics remained unchanged, preserving its established effectiveness. Similarly, the CNN model continued to employ the same strategy for early stopping and learning rate decay, with the learning rate reducing exponentially.

### Advanced Ensemble Strategy with SVM, RF, and XGBoost

While advancing my sentiment analysis model, I realized the need for a more comprehensive approach that could effectively incorporate insights from both text and audio data. To achieve this I used stacking. Stacking[47] is a sophisticated ensemble learning technique that combines multiple classification or regression models via a meta-classifier or a meta-regressor. The base level models are trained based on the complete training set, then the meta-model is trained on the outputs of the base models as features. This approach is beneficial in sentiment analysis, particularly with complex data like song lyrics and audio features, because it leverages the predictive power of multiple learning algorithms, thereby improving overall accuracy. Stacking's unique advantage lies in its ability to blend different kinds of models, capturing diverse patterns and relationships within the data.

XGBoost (eXtreme Gradient Boosting)[48] is a powerful machine learning algorithm that is widely used in winning solutions of various machine learning competitions. It is a scalable and accurate implementation of gradient boosting machines, known for its speed and performance. XGBoost works by building an ensemble of decision trees in a sequential manner, where each tree corrects the errors of its predecessor. This iterative refinement makes XGBoost highly effective in handling complex datasets with multiple features. In my research, XGBoost acts as a meta-classifier to interpret and weigh the predictions of the base models, SVM and RF, which perform best in text and audio analysis, respectively. Its gradient boosting framework allows it to evaluate the relative importance of different features, thus effectively combining textual and audio data insights for sentiment analysis. Because text and audio are two completely different types of data sources, XGBoost may have an advantage in integrating this kind of heterogeneous data, as it can capture complex data patterns and relationships.

### 3.5.3 Combine model Training and analysis

Upon integrating audio features into my sentiment analysis models, I noted that the performance improvement in Dataset 1 was not as significant as expected. This observation could be attributed to the specific nature of Dataset 1, which concentrates on content words in lyrics and their corresponding valence and arousal scores, according to emotion lexicons. The dataset's creation, guided by Russell's model, focused on lyrics that exhibit strong correlations within its four quadrants. This methodological approach might have led to a limited correlation between the labels and the audio features, resulting in a performance that was comparable to the models using only lyrics.

To further validate my approach, I turned to Dataset 2, offering a different perspective. Unlike Dataset 1, Dataset 2 derives from Last.fm user tags, which integrate both audio and lyrical aspects for sentiment tagging. This variation in data origin provides a unique angle to assess the efficacy of combining audio features with textual analysis.

In my initial experiments testing models trained on Dataset 1 on Dataset 2, while overall model performance was not high, ensemble models consistently outperformed those relying solely on the lyrics modality. This result highlights the advantage of a multimodal approach in sentiment analysis.

To further validate the efficacy of my integrated models, I conducted a comparative analysis using Dataset 2 against an established benchmark: an XL-Net-based textual model. This approach involved replicating the structure of the XL-Net model for training and testing on Dataset 2. The results were quite significant. The integrated models, which combined both textual and audio data, not only outperformed their single-modality counterparts (text-only or audio-only models) but also surpassed the benchmark set by the XL-Net model. The findings validated the enhanced effectiveness and generalization capability of the integrated models, underscoring the value of a multimodal approach in music sentiment analysis compared to relying solely on single lyric or audio data types.

The results of all audio feature methodologies will be presented in Chapter 4

#### Dataset 2 analyze

After training the models on Dataset 2, I proceeded to analyze its audio features using PCA, T-SNE, and heatmap visualization, similar to the approach taken with Dataset 1. This step was essential to understand the characteristics and patterns of audio features in Dataset 2, especially

since it incorporates user-generated tags that reflect a combination of lyrical and musical elements in songs.

Based on the 2D and 3D visualizations, it appears that Dataset 2 demonstrates more pronounced clustering tendencies. The use of PCA and T-SNE on Dataset 2 likely contributed to highlighting these distinctive patterns, offering a clear visual representation of how audio features correlate with user-generated emotions. This clustering trend in Dataset2 provides a notable contrast to the insights gained from Dataset1, as referenced in (Figure 3.14) and (Figure 3.15).

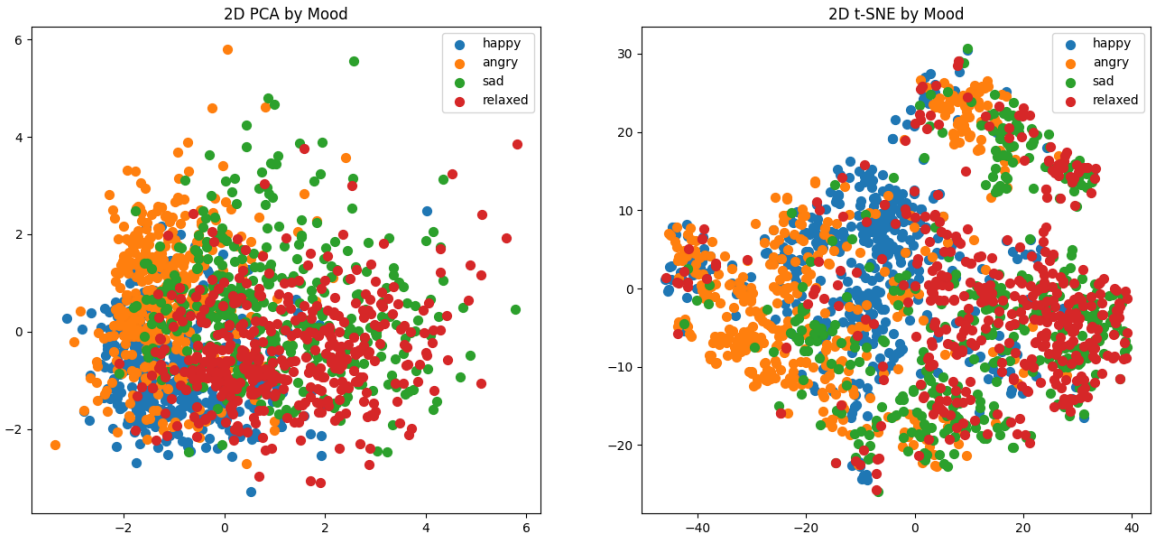


Figure 3.14: 2D PCA and t-SNE in Dataset 2

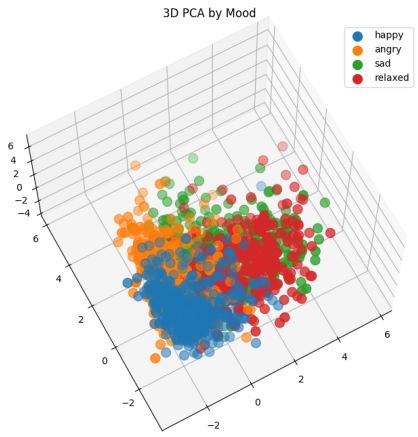


Figure 3.15: 3D PCA in Dataset 2

The heatmap analysis on Dataset 2 focused on revealing the correlations among various audio features. This approach provided an opportunity to uncover how different audio elements interact and contribute to the overall sentiment of songs as perceived by users. Analyzing these correla-

tions after training the models on Dataset 2 helped to pinpoint the specific audio features that played significant roles in shaping the emotional tone of songs in this unique dataset.Furthermore, this analysis revealed that Dataset2 exhibits stronger correlations among its audio features than Dataset1 (Figure 3.16).

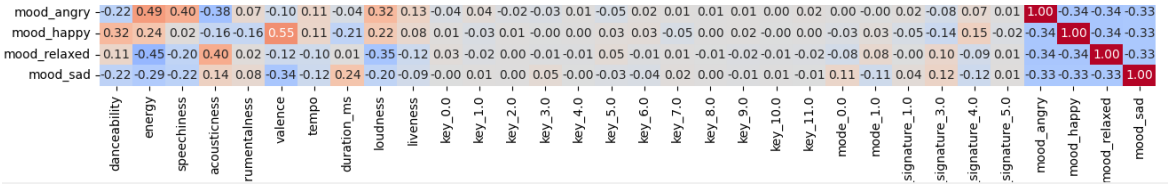


Figure 3.16: Heatmap in Dataset 2

After conducting heatmap analysis on Dataset 2, it became evident that 'Energy' and 'Valence' had the strongest correlation with emotions among the audio features. To delve deeper into this relationship, I created a scatter plot with 'Energy' on one axis and 'Valence' on the other. This visualization was instrumental in understanding the distribution and interaction of these two features in relation to the emotional content of the songs in Dataset 2 (Figure 3.17).

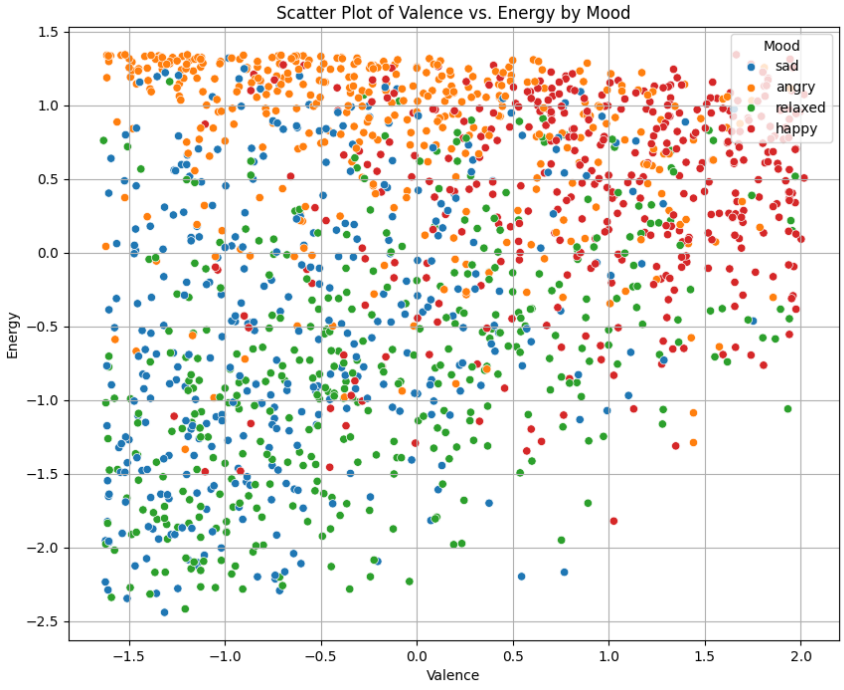


Figure 3.17: V-E in Dataset 2

Analyzing 'Energy' versus 'Valence' in Dataset 2, the patterns in emotional song classification became clear. Tracks in the (-V, +E) quadrant, characterized by low valence and high energy, were predominantly classified as 'Angry'. In contrast, songs in the (+V, +E) quadrant, with

high valence and high energy, tended to be 'Happy'. This trend reflects how high-energy tracks usually evoke intense emotions, with valence indicating whether these emotions are positive or negative.

In the (-V, -E) quadrant, songs with low valence and low energy were primarily labeled as 'Sad' or 'Relaxed', although some overlap was observed between these categories. 'Relaxed' tracks often had marginally higher valence than 'Sad' ones, but the distinction wasn't consistently clear. This indicates that while these audio features suggest emotional tones, they may not always definitively differentiate nuanced states like sadness and relaxation. Incorporating both lyrics and audio data might make this overlap clearer, providing a corrective measure when either modality alone leads to misclassification. This integrative approach can enhance the precision of emotional classification by leveraging the complementary strengths of textual and audio analyses.

This pattern mirrors the labeling approach in both Dataset 1 and Dataset 2, where songs are categorized based on valence and arousal within Russell's model's quadrants. Thus, songs in the (-V, +A) quadrant are 'Angry', those in the (+V, +A) quadrant are 'Happy', those in the (-V, -A) quadrant are 'Sad', and those in the (+V, -A) quadrant are 'Relaxed'. The overlap in textual and audio analysis underscores the value of a multimodal approach in music sentiment analysis, providing a more comprehensive understanding of songs' emotional nuances.

# CHAPTER 4

# Result

## 4.1 Evaluation Metrics

This section presents the comprehensive results of our extensive experiments and analyses, incorporating both textual and audio features in sentiment analysis of song lyrics. These results encapsulate the effectiveness of various methodologies explored throughout the research. Here, we delve into the outcomes achieved by employing different machine learning algorithms and deep learning architectures, and how their performance was influenced by the incorporation of audio features alongside textual data.

### 4.1.1 Confusion Matrix

In the evaluation of our sentiment analysis models, the Confusion Matrix was selected as a key metric due to its effectiveness in four-category emotion classification. This method is especially relevant given that each dataset entry comes with an assigned emotional label, enabling categorization into four distinct emotion classes. The Confusion Matrix provides an in-depth view of classification accuracy across these categories, not only highlighting the quantity of misclassifications but also the nature of these errors[49]. This level of detail is crucial for refining our models and enhancing their performance. Overall, this approach facilitates a thorough assessment of the models' capabilities in interpreting and categorizing various emotional states present in song lyrics and audio features.

| Confusion Matrix | | Predicted Label | | | |
|---|---|---|---|---|---|
| | | **Angry** | **Happy** | **Relaxed** | **Sad** |
| **Ground Truth** | **Angry** | TP | FP | FP | FP |
| | **Happy** | FN | TP | FP | FP |
| | **Relaxed** | FN | FN | TP | FP |
| | **Sad** | FN | FN | FN | TP |

Table 4.1: Confusion Matrix for 4-Category Emotion Classification

In the context of sentiment analysis for song lyrics and audio features, these evaluation measures will be crucial for assessing the effectiveness of the predictive models, as illustrated in the Confusion Matrix (Table 4.1). For our purposes, the terms are defined in relation to the classification of song emotions:

- TP ← Correct identification of a song as belonging to its actual emotional category.
- FP ← Incorrectly labeling a song as belonging to an emotional category it does not.
- FN ← Missing a song that should have been identified as belonging to a specific emotional category.
- TN ← Correctly identifying a song as not belonging to a certain emotional category.

### 4.1.2 Accuracy Score

The Accuracy Score is a fundamental metric in sentiment analysis, measuring the ratio of correctly predicted instances to the total instances. While it provides a quick overview of a model's overall performance, its effectiveness may diminish in imbalanced datasets where one category dominates[50]. In such cases, accuracy might not fully reflect the model's ability to classify all categories equally well. The formula for accuracy is:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

### 4.1.3 Precision Score

Precision Score is an essential metric in sentiment analysis, particularly when the costs of False Positives are high. It measures the proportion of correctly identified positive instances among all instances predicted as positive. This metric is crucial in scenarios where being accurate in the positive predictions is more important than the model's overall accuracy. However, precision

alone doesn't consider False Negatives and might not fully reflect the model's effectiveness across all categories[51]. The formula for precision is:

$$\text{Precision} = \frac{TP}{TP + FP}$$

### 4.1.4 Recall Score

Recall, also known as Sensitivity, is a crucial metric in sentiment analysis, especially when it is vital to capture as many true positives as possible. This metric measures the proportion of actual positives that are correctly identified by the model. It's particularly important when the cost of missing a true positive (False Negative) is high. However, the recall does not account for False Positives, which can be a limitation if specificity is also a concern[51]. The formula for the recall is:

$$\text{Recall} = \frac{TP}{TP + FN}$$

### 4.1.5 F-Beta Score

The F-Beta Score is a valuable metric in sentiment analysis, as it provides a balance between Precision and Recall. It is particularly useful when we need to consider both False Positives and False Negatives in our evaluation. The F-Beta Score becomes the F1 Score when Beta is set to 1, giving equal weight to Precision and Recall. This balanced approach is crucial in situations where both types of errors (False Positives and False Negatives) are equally important[51]. The formula for the F1 Score is:

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

### 4.1.6 Loss Function

In our sentiment analysis task, the dataset is categorized into four distinct sentiment classes: Angry, Happy, Relaxed, and Sad. This clear-cut multiclass classification lends itself well to the use of categorical cross-entropy as the loss function, which is adept at handling situations where each sample is definitively assigned to a single class[52]. Categorical cross-entropy quantifies the divergence between the predicted probability distribution outputted by the model for each class and the actual distribution, represented by one-hot encoded target labels.

$$H(y, \hat{y}) = -\left(y_{\text{Angry}} \log(\hat{y}_{\text{Angry}}) + y_{\text{Happy}} \log(\hat{y}_{\text{Happy}}) + y_{\text{Relaxed}} \log(\hat{y}_{\text{Relaxed}}) + y_{\text{Sad}} \log(\hat{y}_{\text{Sad}})\right) \tag{4.1}$$

In this equation, $y$ is the true label vector represented using one-hot encoding, where the vector has a length equal to the number of classes (four in this case), and the element corresponding to the correct sentiment class is set to 1, with all other elements set to 0. The predicted output $\hat{y}$ from the model is a vector of probabilities, with each element representing the model's predicted probability for each respective class. During training, the goal is to minimize this loss function across all samples in the training dataset, effectively adjusting the model's predictions to align closely with the true labels.

All of the above revealed indicators will be used for quantitative analysis of the results.

## 4.2 Reproduce the results of the paper

| **Model** | **Accuracy (%)** | | **F1 Score (%)** | |
|---|---|---|---|---|
| | **Original** | **Replication** | **Original** | **Replication** |
| NB+Tfidf | 83% | **82%** | 82% | **82%** |
| KNN+Glove | 76% | **71%** | 74% | **70%** |
| SVM+Glove | 71% | **78%** | 68% | **78%** |
| CNN+Glove | 90% | **85%** | 89% | **84%** |
| LSTM+Glove | 90% | **88%** | 90% | **88%** |
| BI-LSTM+Glove | 91% | **88%** | 91% | **88%** |

Table 4.2: Comparative Performance of Original and Replicated Studies on Dataset 1

The replication of the models yielded comparable performance to the original study, indicating a successful validation of the prior work. Notably, the replication of the SVM+Glove model exhibited a significant improvement over the performance reported in the original paper, as shown in Table 4.2. This enhancement not only reaffirms the validity and reliability of the original study as a benchmark but also lays a strong foundation for the subsequent experiments in our research.

## 4.3 Embedding results

| Model + Embedding Method | Preprocessing Combination | Accuracy | F1 Score |
|---|---|---|---|
| BI-LSTM + Glove(Benchmark) | Lemma + LC + NR + SR | 91% | 91% |
| NB + BoW | Lemma + LC + NR + SR | **92%** | **92%** |
| NB + Tf-idf | Lemma + LC + NR + SR | 90% | 90% |
| SVM + Tf-idf | Lemma + LC + NR + SR | **93%** | **93%** |
| SVM + BoW | Lemma + LC + NR + SR | 81% | 82% |
| KNN + Tf-idf | Lemma + LC + NR + SR | 85% | 84% |
| KNN + BoW | Lemma + LC + NR + SR | 69% | 67% |
| CNN + Word2Vec | Lemma + LC + NR + SR | **91%** | **91%** |
| LSTM + Word2Vec | Lemma + LC + NR + SR | 89% | 89% |
| BI-LSTM + Word2Vec | Lemma + LC + NR + SR | **89%** | **89%** |

Table 4.3: Results of Embedding Experiments on Dataset 1

The experimental outcomes post-application of Word2Vec embeddings and after fine-tuning through Grid Search and learning curve analysis demonstrate a notable enhancement in performance across most models compared to the original study that employed GloVe embeddings. The implementation of CNN+Word2Vec achieved the baseline performance set in the original paper, while models like SVM+TFIDF and NB+BOW not only met but exceeded this baseline, as detailed in Table 4.3.

## 4.4 Preprocessing results

| Model + Method | Best Preprocessing Combination | Accuracy (%) | F1 Score (%) |
|---|---|---|---|
| BI-LSTM + Glove(Benchmark) | Lemma + LC + NR + SR | 91% | 91% |
| NB + BoW | Lemma + LC + NR + SR | **92%** | **92%** |
| SVM + Tf-idf | LC + NR + SR | **94%** | **94%** |
| CNN + Word2Vec | LC + NR + SR | **92%** | **92%** |
| BI-LSTM + Word2Vec | Lemma + NR + SR | 90% | 90% |

Table 4.4: Results of Preprocessing Experiments on Dataset 1

The results in Table 4.4 show significant improvements in model performance due to customized preprocessing strategies. Optimizations like GridSearch and learning curve analysis have enabled the fine-tuning of preprocessing for each model. Notably, the SVM model has shown remarkable enhancement, as evidenced in Figure 4.1 and Figure 4.2, which illustrate the learning curves before and after preprocessing. These visualizations highlight advancements in performance and validation accuracy, affirming the impact of precise preprocessing in sentiment analysis.

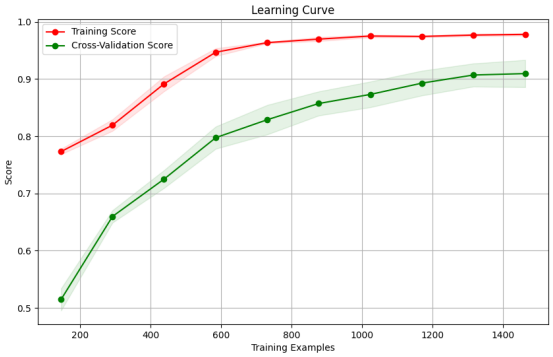


Figure 4.1: Learning curve of SVM before the best preprocessing

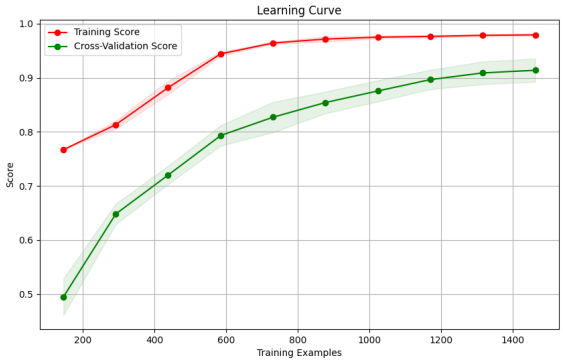


Figure 4.2: Learning curve of SVM the best preprocessing

## 4.5 Lrycis only Model architecture and parameters

The parameter settings for each model after text lyric processing are as follows:

### NB+BoW

The model utilizes a MultinomialNB classifier with an alpha parameter of 0.5 to handle frequency data and smooth probability calculations. Text is transformed into a bag-of-words format using the CountVectorizer, with parameter settings of a maximum document frequency of 0.5, a minimum document frequency of 3, and an n-gram range of (1, 2). This configuration helps capture both single words and bi-gram combinations, thereby better understanding the content of the text.

### SVM+Tf-idf

This model employs a linear kernel SVM for text classification. The regularization parameter C is set to 0.36, balancing model complexity and accuracy. Text data is processed through Tf-idfvectorization, where the TfidfVectorizer's parameters include a maximum document frequency of 0.5, a minimum document frequency of 0.001, and an n-gram range of (1, 1). These settings aim to maximize the utility of word frequency information while mitigating the impact of overly common or rare words.

### CNN+Word2Vec

The CNN model is structured to process input sequences of length 250. It consists of three convolutional layers, each followed by a max-pooling layer and a dropout layer, which are included

to mitigate the risk of overfitting. The initial layer of the model, an embedding layer, is responsible for mapping word indices into a 300-dimensional space using pre-trained word vectors. The convolutional layers are designed with kernel counts of 128, 64, and 32, respectively, and each has a kernel size of 5. Following each convolutional layer, there are max-pooling and dropout layers. The network also includes a global max-pooling layer that connects to a fully connected layer, which is used for the classification task. The training of the model is set up with parameters including 20 epochs, with early stopping set to 20, a batch size of 16, and an initial learning rate of 0.0005, which undergoes exponential decay at a rate of 0.1.

#### Bilstm+Word2Vec

The input layer processes sequences of length 250. An embedding layer generates word embedding vectors, followed by a dropout layer with a 20% ratio. The bidirectional LSTM layer consists of 100 hidden units, reading the input from both directions and merging their outputs. Finally, a fully connected layer with a Softmax activation function is used for multi-class classification. Training parameters are set for 30 epochs with batches of 16 samples each, using an Adam optimizer with a learning rate of 0.0001.

## 4.6 Audio Feature Results

#### Dataset 1 Audio Only

| Model | Accuracy (%) | F1 Score(%) |
|---|---|---|
| SVM | 39% | 36% |
| Desen | 38% | 37% |
| RF | 40% | 40% |

Table 4.5: Audio Only Model Performance in Dataset 1

The results in 4.5 show that models relying only on audio features perform poorly on Dataset 1.

#### Dataset 1 Train and Test with Lyrics and Audio Feature

| Model | Accuracy (%) | F1 Score (%) |
|---|---|---|
| SVM + Tf-idf (Early Fusion) | 89% | 89% |
| CNN + DenseNet+Word2Vec | **92%** | **92%** |
| BiLSTM + DenseNet+Word2Vec | 90% | 90% |
| Ensemble(Stacking) | 91% | 91% |

Table 4.6: Performance comparison of combine models on Dataset 1

Table 4.6 shows the performance comparison of various models on Dataset 1 when trained and tested with both lyrics and audio features. In our analysis of these results, we observed that integrating audio features did not substantially improve model performance, likely due to the dataset's exclusive focus on lyric content with strong emotional sentiments. Notably, among the models tested, CNN + DenseNet+Word2Vec demonstrated better performance, suggesting some models may be more adept at leveraging audio features. Consequently, it becomes essential to evaluate the generalization of audio features in varying contexts. Dataset 2, with its user-centric perspective, provides a contrasting setting to Dataset 1. Here is the learning curve (Figure 4.3), loss curve (Figure 4.4), and architecture (Figure 4.5) of CNN + DenseNet+Word2Vec.

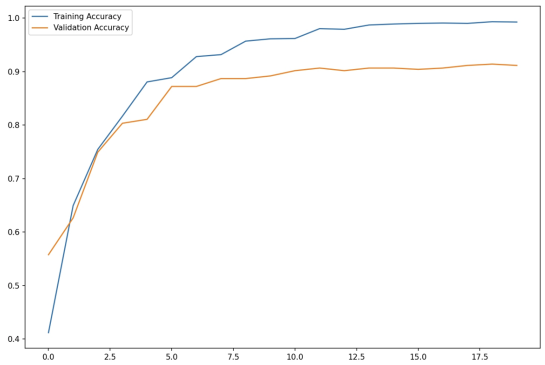


Figure 4.3: Learning curve of CNN + DenseNet+Word2Vec

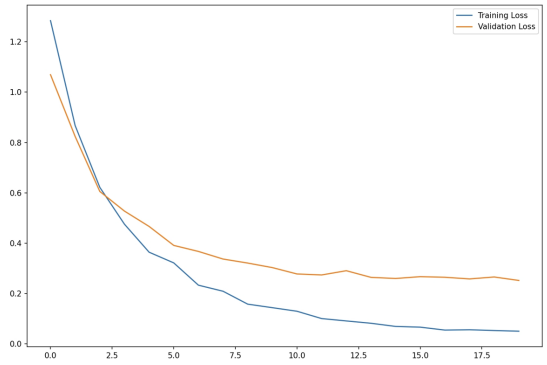


Figure 4.4: Loss curve of CNN + DenseNet+Word2Vec

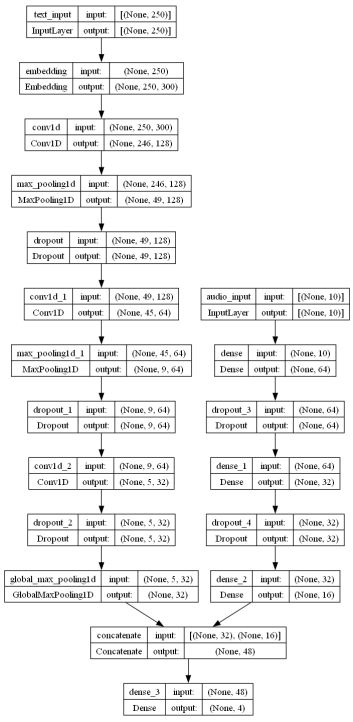


Figure 4.5: Architecture of CNN + DenseNet+Word2Vec

**Dataset 2 Test with Dataset 1 model**

| Method | Configuration | Overall F1 Score |
|---|---|---|
| SVM+Tf-idf | Lyrics Only | 32% |
| SVM+Tf-idf | Combined | **34%** |
| CNN+Word2Vec | Lyrics Only | 36% (Angry F1: 49) |
| CNN+DenseNet+Word2Vec | Combined | **38% (Angry F1: 54)** |
| Bi-LSTM+Word2Vec | Lyrics Only | 35% |
| BI-LSTM+DenseNet+Word2Vec | Combined | **37%** |
| NB+BoW | Lyrics Only | 35% |
| Ensemble(Stacking) | Combined | **37%** |

Table 4.7: Comparison of F1 Scores across Different Models and Configurations (Trained on Dataset 1 and Tested on Dataset 2)

Based on the results in Table 4.7 , we observed that despite generally low F1 scores, models that utilized a combination of lyrics and audio features outperformed those that relied solely on lyrics. This trend was particularly notable in the CNN model, which displayed significant improvements. To further affirm the value of audio features, we trained and tested using Dataset 2 with the same model parameters and architecture. In addition, a study that employed an XL-Net model focused solely on the lyrics from Dataset 2 was used as a benchmark for comparison. This methodical approach aimed to provide a comprehensive assessment of the models' performance, reinforcing the relevance of audio features in the classification of text and audio data.

**Dataset 2 Train and Test**

| Model | Configuration | F1 Score |
|---|---|---|
| XL-Net + Lemma Benchmark | Lyrics Only | 59% |
| SVM+Tf-idf | Lyrics Only | 54% |
| SVM+Tf-idf | Combined | **62%** |
| CNN+Word2Vec | Lyrics Only | 57% |
| CNN+DesenNet+Word2Vec | Combined | **68%** |
| Ensemble(Stacking) | Combined | **64%** |
| NB+BoW | Lyrics Only | 52% |
| BI-LSTM+Word2Vec | Lyrics Only | 53% |
| BI-LSTM+DesenNet+Word2Vec | Combined | **64%** |
| SVM | Audio Only | 61% |

Table 4.8: Comparison of F1 Scores across Different Models and Configurations (Trained and Tested on Dataset 2)

Based on the results in Table 4.8, it was found that the composite models exhibited better performance in Dataset 2 compared to models using either lyrics or audio features in isolation. Notably, the CNN+DenseNet+Word2Vec achieved an F1 score of 68%, which is significantly

higher than the baseline of 59% reported in the reference paper and surpasses the performance of both other individual and composite models.

## 4.7 Combine model architecture and parameters

### SVM + Tf-idf (Early Fusion)

For the SVM + Tf-idf (Early Fusion), the same word embedding with Tf-idf was utilized with the addition of the regularization parameter C set to 0.36 to balance the complexity and accuracy of the model.

### CNN and Bi-LSTM models

For both CNN and Bi-LSTM models, the text processing components remained unchanged from their previous configurations. In parallel, DenseNet was utilized to process audio features, with a shared architecture across both models. This architecture consists of an audio input layer followed by three Dense layers with 64, 32, and 16 units, respectively, each with ReLU activation for hierarchical feature extraction. Dropout layers with a rate of 0.2 are interleaved between Dense layers to reduce overfitting. The processed audio features are then merged with the text feature layers. The combined features feed into a Dense classifier layer with a softmax activation function corresponding to the number of output labels. For the CNN model, early stopping is set to 20 epochs, and an exponential decay rate of 0.1 is applied. The models are compiled using the categorical_crossentropy loss function, optimized with Adam, and track accuracy as the performance metric, offering a robust framework for evaluating the synergistic effect of text and audio data.

### Stacking

For stacking, following rigorous optimization with Grindsreach, our model employs a layered approach for comprehensive data analysis.The SVM component, designated for text data processing, maintains consistent parameters and architecture as established in previous text handling iterations.In parallel, for the audio data, a RF Classifier with n_estimators set to 100 has been optimized for maximum efficiency in interpreting audio features.Central to the architecture of our model is the XGBoost Classifier. Configured with 100 estimators, a learning rate of 0.4, a maximum depth of 3, and an objective set to 'multi:softmax', the XGBoost Classifier is adeptly suited for the nuances of our multiclass classification task.

# CHAPTER 5

# Evaluation and Discussion

## 5.1 Lyrcis Evaluation

In evaluating the results of the replication of my research, we validated the authority of the baseline paper and discovered limitations of GloVe embeddings related to deep learning's max_sequence and the preprocessing of lyric texts. One key limitation we identified was the global focus of GloVe, which often does not align well with the intricate requirements of processing lyric texts. Subsequent experiments indicated that Word2Vec, Tf-idf, and BoW were significantly more effective than GloVe in handling lyric texts. This superior performance is largely due to their emphasis on context and the importance of individual words. This approach more effectively aligns with the unique attributes of musical lyrics, which are characterized by rhythmic and repetitive patterns[37][38]. These findings highlight the effectiveness of embeddings that focus on contextual understanding and word-level significance, in contrast to GloVe's broader, more global approach to text representation.

In preprocessing, particularly for lyrics, it was observed that after preprocessing, 97% of the lyrics in both Dataset 1 and Dataset 2 fall within a 0-250 sequence length range. Establishing this specific length was crucial in minimizing the noise from max_sequence 0-padding, thereby enhancing the model's performance [53]. Our research also found that various models perform differently to distinct preprocessing methods. The widespread use of abbreviations and slang in lyrics necessitated steps like NR and SR are required for each model, which significantly improved both the accuracy and generalizability of the models. However, stem, which is designed to reduce

words to their root form, was less effective. This lessened effectiveness can largely be attributed to stem's potential to alter the original meaning or context of words in lyrics[54], a vital aspect for precise sentiment analysis.

After comparing the performance of different models in the task of sentiment classification of song lyrics, I observed that SVM and CNN models perform better without Lemma preprocessing, while NB and Bi-LSTM models show improved performance with Lemma preprocessing. This finding closely relates to the inherent processing mechanisms of these models and their sensitivity to lexical variations.

Particularly, the combination of NB with the BoW model and Bi-LSTM with Word2Vec both exhibited high sensitivity to word form changes. NB, relying on statistical analysis of word frequencies, benefits from Lemma preprocessing as it consolidates different forms of the same word, thus enhancing frequency estimation accuracy, crucial for capturing subtle emotional nuances in lyrics. Concurrently, Bi-LSTM leverages its long-term dependency capturing ability with Word2Vec's rich semantic information to more accurately interpret emotions in lyrics. Lemma preprocessing provides a more uniform word base form, aiding the model in better understanding emotional coherence[55].

Conversely, models combining SVM+Tf-idf and CNN+Word2Vec demonstrate strong robustness to lexical form variations. SVM effectively differentiates between emotional categories by constructing optimal boundaries, not overly relying on individual word forms[56]. Simultaneously, CNN identifies key patterns and features from Word2Vec's embeddings[35], often transcending word surface forms, rendering Lemma preprocessing less impactful on CNN's performance.

Notably, while both CNN and Bi-LSTM utilize Word2Vec embeddings, their approaches to processing these embeddings are distinct. CNN focuses on capturing immediate, local features within the embeddings, excelling in identifying specific emotional indicators in the text[35]. This capability is particularly effective in handling song lyrics, as the rhythmic and repetitive nature of lyrics aligns well with CNN's strength in identifying and interpreting local patterns. On the other hand, Bi-LSTM, with its ability to understand context over longer text spans, benefits more from the consistent word forms offered by lemma preprocessing, utilizing Word2Vec's depth to grasp the overarching emotional narrative in lyrics[36]. However, according to the results, CNN is better at handling lyrical content.

Finally, In the single lyric experiment,the combination of optimal preprocessing improves the accuracy and verification set by 1%, the SVM model combined with Tf-idf achieved a high accuracy rate of 94%, surpassing the baseline set in the study. This result also revealed that SVM and NB, using BoW and Tf-idf embeddings, performed better on this dataset compared to

other deep learning models, possibly due to the high emotional weight of specific words in the lyrics. In certain emotional categories, the frequency of some keywords is notably higher than in others, making these words significant indicators of emotion[57]. For instance, as shown in Table 5.1, in the "happy" category, the word "love" carries substantial weight. For the BoW method, which is based on word frequency, this means these words play a decisive role in classification.

Simultaneously, when using the Tf-idf method, these words are not only significant due to their high frequency in specific categories but also gain higher weight(idf) due to their uniqueness across the entire dataset. This approach helps emphasize words that are significant to particular emotional categories but less common in other parts of the dataset. Therefore, the specific distribution of these emotionally charged words provides a powerful distinguishing feature for models employing these embedding methods, thereby enhancing the accuracy of emotion classification.

| **Relaxed** | **Happy** | **Angry** | **Sad** |
|---|---|---|---|
| home (1266) | love (9535) | fire (1153) | lonely (736) |
| baby (1019) | baby (1244) | war (653) | time (560) |
| girl (1010) | know (1077) | like (602) | know (511) |
| love (724) | oh (1053) | know (557) | like (452) |
| oh (717) | like (754) | oh (520) | get (422) |

Table 5.1: Top words per mood category with their counts in Dataset1

## 5.2 Audio and Combine model Evaluation

Regarding audio features, the results during the feature selection stage revealed important considerations regarding the impact of each audio feature on emotions. This can reduce the curse of dimensionality[58], allowing the model to focus more on audio features relevant to the emotional dimension, thereby enhancing model performance and generalization. In the analysis of individual audio features' emotional dimensions, their impact was relatively minor within Dataset 1, where labels showed a strong correlation with the lyrics. This is largely associated with Dataset 1's focus on lyrical emotional dimensions. However, Dataset 1's visualizations and heatmaps did reveal some correlations with audio features, but including these features in the composite model did not significantly enhance accuracy in Dataset 1 tests. Performance-wise, CNN+Densenet+Word2Vec was on par with the lyrics only modality CNN+Word2Vec. Due to observations that the validation set loss and performance did not converge well without exponential decay, tending to overfit quite early, early stopping and exponential decay methods were employed. Comparatively, models using early stopping and exponential decay(Figure 5.2) showed smoother and better-fitting training curves than those without exponential decay(Figure 5.1). This method contributes to more

efficient training and potentially better generalization capabilities by focusing on the model's learning process and preventing overfitting.

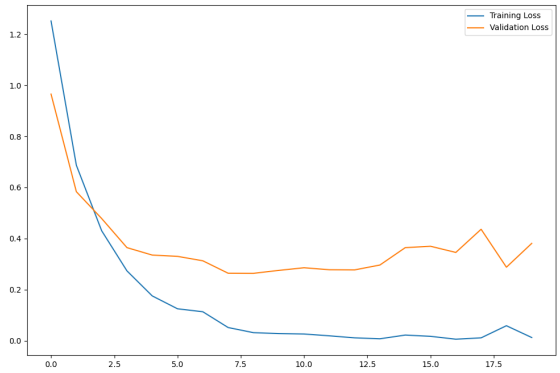


Figure 5.1: CNN Model without Decay

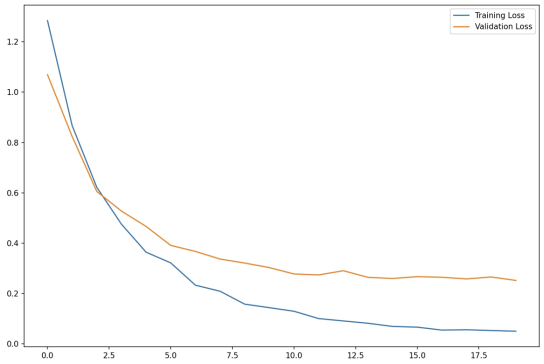


Figure 5.2: CNN Model with Decay

Therefore, when further considering the generalizability of the model, introducing a second, more diverse music dataset provided additional insights. Despite lower overall results when testing the model trained on Dataset 1 with Dataset 2, primarily due to 7,775 unseen tokens in Dataset 2 causing out-of-vocabulary issues, the composite model exhibited better generalization capabilities compared to single-modality models. The assessment of results indicated that SVM+Tf-idf performed better in the lyric only modality than CNN+Word2Vec, but CNN+Word2Vec showed stronger generalization. The higher performance of the composite model observed in the tests may be attributed to findings from Dataset 1's Heatmaps. These heatmaps indicated a strong positive correlation between 'Angry' emotions and audio features such as Energy and Loudness, and a negative correlation with the 'Relaxed' emotion." This pattern suggests that songs with higher Energy and Loudness are indicative of "Angry," while those with lower Loudness represent "Relaxed." This was validated when the composite models and lyrics only models trained on Dataset 1 were tested on Dataset 2. Here, the composite model CNN+Densenet+Word2Vec showed a 5% improvement in F1 score compared to the single-modality CNN+Word2Vec, proving that the models trained on Dataset 1 had learned this correlation, even though it was not as strong in Dataset 1. Compared to Dataset 1, the overall audio feature visualizations and heatmaps in Dataset 2 also showed stronger patterns and linear correlations, indicating that these user-subjective emotion labels in Dataset 2 were more sensitive and significant to audio features. Compared to Dataset 1, the overall audio feature visualizations and Heatmaps in Dataset 2 also showed stronger patterns and linear correlations, indicating that in Dataset 2, this user-subjective perspective emotional labels are more sensitive and significant to audio features than in Dataset 1.

To further validate the proposed method, I conducted a study based on literature using Dataset 2 as a benchmark, training and testing the same model architecture and parameters on this dataset. The results showed that the proposed composite model significantly outperformed the single-modality models, exceeding the baseline established in the referenced paper. The CNN+DenseNet+Word2Vec model achieved a 68% F1 score. This success underscores a key discovery: Russell's model quadrants can effectively map emotions in lyrics and audio features. It was observed that in lyric classification, the quadrant (-V, +A) corresponds to "Angry", (+V, +A) to "Happy", (-V, -A) to "Sad", and (+V, -A) to "Relaxed". Similarly, in my audio analysis, (-V, +E) corresponds to "Angry", (+V, +E) to "Happy", (-V, -E) to "Sad", and (+V, -E) to "Relaxed".These results validate the effectiveness of incorporating Energy (E) alongside Arousal (A) in Russell's model, offering a nuanced understanding of the emotions in music.

This alignment between the emotion classification methods for music and the V-A and V-E model quadrants demonstrates a significant parallel in how both textual and audio features capture the emotional essence of a song. The finding confirms that the underlying patterns in the textual and audio features, as delineated by the Russell model, can more accurately define a song's emotional context.

This relationship underscores the advantages of integrating textual and audio features to enhance the model's generalizability and accuracy. By leveraging the strengths of both modalities, the approach provides a comprehensive and nuanced understanding of a song's emotional landscape, as conceptualized by the V-A and V-E models. This integrative method proves particularly effective in capturing the complex emotional expressions inherent in music.

## 5.3 Use case Evaluation

After obtaining these results, I further assessed the model's effectiveness by applying it to analyze the emotions of the top 100 Spotify songs for each year from 2013 to 2023, linking the findings with historical events to demonstrate the model's wide applicability. This analysis provides a reliable dataset for music emotion analysis, a field currently hampered by the lack of publicly available datasets that simultaneously consider both lyrics and audio, primarily due to copyright issues. Through cross-validation and accuracy analysis, the CNN model exhibited the best average generalization capability and precision across both Dataset 1 and Dataset 2, leading to its selection for this task.

The results in Figure 5.3 revealed two parallel trends from 2020 to 2022 in the emotional content of top Spotify songs. First, there was an increase in the prevalence of "sad" songs,

peaking in 2022. This uptick might reflect a collective gravitation towards more introspective and contemplative themes in music, potentially mirroring the public mood during these years. Alongside this, I observed a marked decrease in "happy" songs, dropping to the lowest levels seen in the last five years. This trend could indicate a reduced production or popularity of more upbeat, optimistic musical themes, perhaps in response to the global atmosphere of the time, influenced by significant societal and economic challenges.

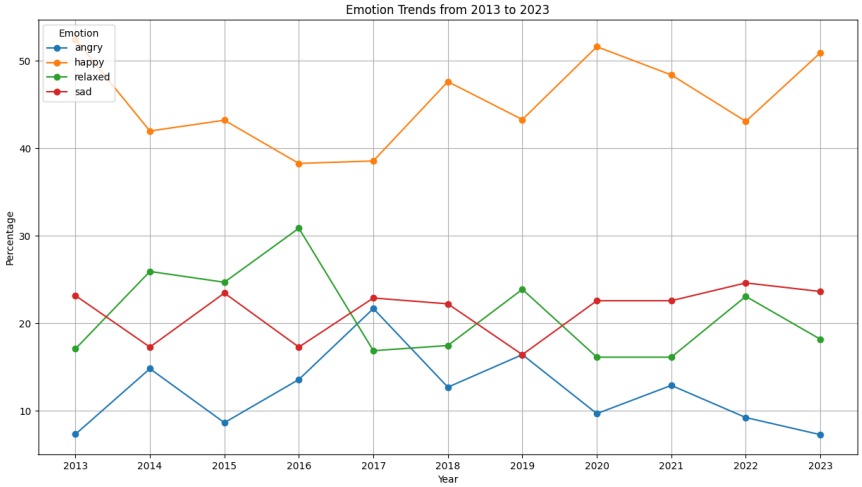


Figure 5.3: Predictive Trends of Spotify Top 100 Tracks from 2013 to 2023

The investigation into authoritative literature from 2020 to 2022 validates the effectiveness and reliability of my model and dataset, especially concerning societal events of this period, including the COVID-19 pandemic from 2020 to 2022 and the Russia-Ukraine conflict in 2022. These external events have indirectly substantiated the trends observed in my model's predictions.

A large-scale survey during the COVID-19 pandemic reported by Neuroscience News[59] revealed that individuals experiencing increased negative emotions used music for emotional regulation, while those with positive feelings turned to music as a social interaction substitute. My model aligns with these findings, showing an increase in sad songs and a decrease in happy songs during the pandemic to meet the evolving socio-emotional needs.Moreover, the study 'Music and mood regulation during the early stages of the COVID-19 pandemic'[60] found that during the quarantine period of the pandemic, individuals feeling stressed or sad could improve their mood by listening to negatively-valanced music. This indicates that there is a strong connection between emotional states and music preferences during periods of heightened stress and isolation, particularly in times of heightened stress and isolation. Another study highlighted by Denk et al. [61] emphasizes the COVID-19 pandemic's significant impact on overall music market consumption and consumer spending, which could have influenced the emotional content of music

produced during this period.

The Russia-Ukraine conflict of 2022, occurring concurrently with the COVID-19 pandemic, has had a substantial impact on mental health on a global scale. Studies and reports, including those from the United Nations, highlight the dual burden of the conflict and the pandemic, exacerbating mental health conditions such as depression, anxiety, and stress[62]. Notably, the increase in symptoms like war anxiety among large segments of the population indicates the widespread and severe emotional impact of these concurrent crises[63]. The conflict has not only disrupted healthcare systems and compounded psychological distress in Ukraine, where there was a pre-existing burden of communicable diseases and limited medical supplies, but its effects have also extended globally, aggravating the distress caused by the ongoing pandemic[64].

Together, these insights highlight how societal events like the COVID-19 pandemic and the Russia-Ukraine conflict from 2020 to 2022 profoundly affected individual emotional states and coping strategies, subsequently influencing their music preferences. The trends observed in my model, particularly the peak in sad songs in 2022, are substantiated by these studies. This correlation emphasizes the model's ability to capture subtle changes in music preferences during times of global stress and uncertainty, offering a nuanced understanding of how external events can impact cultural and personal expression.

# CHAPTER 6

# Future Work and Limitation

## 6.1 Selection of Audio Features

In my research, I utilized Heatmaps and RF to assess the importance and relevance of certain audio features. However, this approach may not have fully considered the impact of features like key, mode, and time signature on emotional analysis, leading me to exclude these features. Reflecting on this, future work should delve deeper into analyzing these audio features to determine whether they indeed negatively impact model performance or if they could be significant in certain emotional expressions or musical genres. Expanding this investigation, new statistical methods and machine learning models could more accurately determine the value and role of these features in various contexts, thereby making the selection of audio features more scientific and comprehensive.

## 6.2 Dataset Enhancement

Regarding lyric data processing, building upon the current approach, the size of Datasets 1 and 2, along with the richness of the vocabulary, might influence model performance. Specifically addressing this, future work could involve expanding the vocabulary, including words less prevalent in the existing dataset, to enhance the model's understanding and processing capabilities for various lyric types. This enhancement would improve the model's ability to handle complex textual content, especially lyrics containing uncommon or culturally specific vocabulary, thereby

enhancing the model's generalization ability and accuracy.

## 6.3 Confidence Analysis in Text and Audio Modalities

Considering the integration of modalities, in my research, when integrating the text and audio modalities in the composite model, the confidence score between modalities was not considered. To remedy this, future work could involve independently analyzing the confidence of each modality for different emotional categories, particularly since angry might be more prominently represented in audio features, while sad might be more easily recognized in text features. Strengthening the modal fusion strategy, using technologies such as attention mechanisms before merging, could evaluate the contribution of each modality more effectively. Moreover, employing a weighted average method to combine the confidence scores of both modalities, with weights based on the reliability of predictions for specific emotional categories, could be explored. Additionally, developing a collaborative learning strategy would enable the model to simultaneously learn the interrelations between text and audio data and their performance in different emotional dimensions.

## 6.4 Causal Relationship between Text and Audio Data

Lastly, in my research, the model still processes text and audio data independently to some extent, and does not explicitly address their dynamic relationship or interaction before combining these data. Therefore, it does not consider the dependency relationship between text and audio. Future work in this area could consider undertaking combined feature engineering to analyze how specific types of lyrics (such as Happy or Sad) usually combine with certain audio features (like Danceability, Loudness). Additionally, further work could also involve acquiring pure audio signals of music as time series data and using temporal models to process these audio data, to capture the emotional dynamics changing over time in music. These future models could better understand the dependency relationship between text and audio.

# CHAPTER 7

# Conclusion

In conclusion, this research has effectively analyzed lyrics and Spotify audio features to deeply understand the intricate emotional content within musical compositions. Not only did it successfully replicate a method from a key paper in the field of MER, but it also made significant improvements. This research employed text embedding techniques that focus more on word frequency and context, combined with advanced preprocessing and machine learning models. The improvements in emotional analysis have been substantiated by the significant F1 scores achieved. Integrating Spotify audio features into the model significantly enhanced its generalizability, and a dataset created using this model has been validated with real-world societal data. This provides a new perspective for music emotion analysis, demonstrating a more detailed and comprehensive approach. The insights gained from this research not only enhance our understanding of emotional expression in music but also pave the way for future research in the field, potentially leading to more personalized and emotionally resonant musical experiences. Thus, this work demonstrates the immense potential of combining various data sources and advanced analytical techniques to decode the complex emotional tapestry of music.

# CHAPTER 8

# Appendices

# Bibliography


[1] Patrik N Juslin, John A Sloboda, et al. Music and emotion. *D. DEUTSCH (Org.)*, 2001.

[2] Malcolm Budd. *Music and the emotions: The philosophical theories*. Routledge, 2002.

[3] Piotr Przybysz. Music and emotions. *AVANT. Pismo Awangardy Filozoficzno-Naukowej*, (3):174–196, 2013.

[4] Marcel Zentner, Didier Grandjean, and Klaus R. Scherer. Emotions evoked by the sound of music: characterization, classification, and measurement. *Emotion*, 8 4:494–521, 2008.

[5] Shuai-Ting Lin, Pinchen Yang, Chien-Yu Lai, Yu-Yun Su, Yi-Chun Yeh, Mei-Feng Huang, and Cheng-Chung Chen. Mental health implications of music: Insight from neuroscientific and clinical studies. *Harvard review of psychiatry*, 19(1):34–46, 2011.

[6] Youngmoo E Kim, Erik M Schmidt, Raymond Migneco, Brandon G Morton, Patrick Richardson, Jeffrey Scott, Jacquelin A Speck, and Douglas Turnbull. Music emotion recognition: A state of the art review. In *Proc. ismir*, volume 86, pages 937–952, 2010.

[7] Tao Li and Mitsunori Ogihara. Detecting emotion in music. In *International Society for Music Information Retrieval Conference*, 2003.

[8] Audrey Laplante. Users' relevance criteria in music retrieval in everyday life: An exploratory study. In *International Society for Music Information Retrieval Conference*, 2010.

[9] Jinhyeok Yang, Woo-Joon Chae, SunYeob Kim, and Hyebong Choi. Emotion-aware music recommendation. In *Interacción*, 2016.

[10] J Stephen Downie. Music information retrieval. *Annual review of information science and technology*, 37(1):295–340, 2003.

[11] Alf Gabrielsson & Lindström and Erik. The role of structure in the musical expression of emotions. In Patrik N. Juslin and John Sloboda, editors, *Handbook of Music and Emotion: Theory, Research, Applications*. Oxford University Press, 2011.

[12] Kate Hevner. Experimental studies of the elements of expression in music. *The American journal of psychology*, 48(2):246–268, 1936.

[13] James A Russell. A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161, 1980.

[14] Ulrich Schimmack and Alexander Grob. Dimensional models of core affect: a quantitative comparison by means of structural equation modeling. *European Journal of Personality*, 14:325 – 345, 2000.

[15] Juan Sebastián Gómez Cañón, Estefanía Cano, Tuomas Eerola, Perfecto Herrera, Xiao Hu, Yi-Hsuan Yang, and Emilia Gómez. Music emotion recognition: Toward new, robust standards in personalized and context-sensitive applications. *IEEE Signal Processing Magazine*, 38:106–114, 2021.

[16] Yi-Hsuan Yang, Yu-Ching Lin, Ya-Fan Su, and Homer H. Chen. A regression approach to music emotion recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(2):448–457, 2008.

[17] Yuan-Pin Lin, Yi-Hsuan Yang, and Tzyy-Ping Jung. Fusion of electroencephalographic dynamics and musical contents for estimating emotional responses in music listening. *Frontiers in neuroscience*, 8:83280, 2014.

[18] Yisi Liu, Olga Sourina, and Minh Khoa Nguyen. Real-time eeg-based emotion recognition and its applications. *Transactions on Computational Science XII: Special Issue on Cyberworlds*, pages 256–277, 2011.

[19] Mireille Besson, Frederique Faita, Isabelle Peretz, A-M Bonnel, and Jean Requin. Singing in the brain: Independence of lyrics and tunes. *Psychological Science*, 9(6):494–498, 1998.

[20] Rudolf Mayer and Andreas Rauber. Musical genre classification by ensembles of audio and lyrics features. In *Proceedings of international conference on music information retrieval*, pages 675–680, 2011.

[21] Michael Fell and Caroline Sporleder. Lyrics-based analysis and classification of music. In *Proceedings of COLING 2014, the 25th international conference on computational linguistics: Technical papers*, pages 620–631, 2014.

[22] Erion Çano and Maurizio Morisio. Moodylyrics: A sentiment annotated lyrics dataset. In *Proceedings of the 2017 international conference on intelligent systems, metaheuristics & swarm intelligence*, pages 118–124, 2017.

[23] Jiddy Abdillah, Ibnu Asror, Yanuar Firdaus Arie Wibowo, et al. Emotion classification of song lyrics using bidirectional lstm method with glove word representation weighting. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 4(4):723–729, 2020.

[24] Erion Çano, Maurizio Morisio, et al. Music mood dataset creation based on last. fm tags. In *2017 International Conference on Artificial Intelligence and Applications, Vienna, Austria*, pages 15–26, 2017.

[25] Yinan Zhou. *Music Emotion Recognition on Lyrics Using Natural Language Processing*. McGill University (Canada), 2022.

[26] Hande Aka Uymaz and Senem Kumova Metin. Vector based sentiment and emotion analysis from text: A survey. *Engineering Applications of Artificial Intelligence*, 113:104922, 2022.

[27] Samar Al-Saqqa and Arafat A. Awajan. The use of word2vec model in sentiment analysis: A survey. *Proceedings of the 2019 International Conference on Artificial Intelligence, Robotics and Control*, 2019.

[28] Yash Sharma, Gaurav Agrawal, Pooja Jain, and Tapan Kumar. Vector representation of words for sentiment analysis using glove. *2017 International Conference on Intelligent Communication and Computational Techniques (ICCT)*, pages 279–284, 2017.

[29] Marvin Ray Dalida, Lyah Bianca Aquino, William Cris Hod, Rachelle Ann Agapor, Shekinah Lor Huyo-a, and Gabriel Avelino Sampedro. Music mood prediction based on spotify's audio features using logistic regression. In *2022 IEEE 14th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, pages 1–5. IEEE, 2022.

[30] RUSSELL'S CIRCUMPLEX MODEL and VECTOR DISTANCE CALCULATION TO. International journal of modern pharmaceutical research. *Psychology*, 11:14.

[31] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[32] Thomas Bayes. Lii. an essay towards solving a problem in the doctrine of chances. by the late rev. mr. bayes, frs communicated by mr. price, in a letter to john canton, amfr s. *Philosophical transactions of the Royal Society of London*, (53):370–418, 1763.

[33] Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27, 1967.

[34] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152, 1992.

[35] Ye Zhang and Byron Wallace. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *arXiv preprint arXiv:1510.03820*, 2015.

[36] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[37] Gulnoza Murodilovna Oripova. Rhythm and mything in lyrical genre. , 2020.

[38] Kento Watanabe, Yuichiroh Matsubayashi, Naho Orita, Naoaki Okazaki, Kentaro Inui, Satoru Fukayama, Tomoyasu Nakano, Jordan Smith, and Masataka Goto. Modeling discourse segments in lyrics using repeated patterns. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1959–1969, 2016.

[39] Zellig S. Harris. Distributional structure. *WORD*, 10(2-3):146–162, 1954.

[40] Karen Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21, 1972.

[41] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.

[42] BRIAN HOPKINS and J. G. SKELLAM. A New Method for determining the Type of Distribution of Plant Individuals. *Annals of Botany*, 18(2):213–227, 04 1954.

[43] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

[44] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[45] Leland Wilkinson and Michael Friendly. The history of the cluster heat map. *The American Statistician*, 63(2):179–184, 2009.

[46] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[47] David H Wolpert. Stacked generalization. *Neural networks*, 5(2):241–259, 1992.

[48] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[49] Richard O Duda, Peter E Hart, et al. *Pattern classification and scene analysis*, volume 3. Wiley New York, 1973.

[50] Haibo He and Edwardo A Garcia. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9):1263–1284, 2009.

[51] CJ van Rijsbergen. *Information retrieval*. Butterworth-Heinemann, 1979.

[52] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[53] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27, 2014.

[54] Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to information retrieval*. Cambridge university press, 2008.

[55] Constituency Parsing. Speech and language processing. *Power Point Slides*, 2009.

[56] Thorsten Joachims. Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning*, pages 137–142. Springer, 1998.

[57] Bing Liu. *Sentiment analysis and opinion mining*. Springer Nature, 2022.

[58] Richard Bellman. Dynamic programming. *science*, 153(3731):34–37, 1966.

[59] Neuroscience News. Music can help reduce anxiety and stress during covid, 2021. Accessed: insert-date-of-access.

[60] Sarah Hennessy, Matthew Sachs, Jonas Kaplan, and Assal Habibi. Music and mood regulation during the early stages of the covid-19 pandemic. *PLOS ONE*, 16(10):e0258027, 2021.

[61] Janis Denk, Alexa Burmester, Michael Kandziora, and Michel Clement. The impact of covid-19 on music consumption and music spending. *PLOS ONE*, 17(5):e0267640, 2022.

[62] United Nations. The human toll and humanitarian crisis of the russia-ukraine war: the first 162 days. *BMJ Global Health*, 2022.

[63] BMJ Global Health. Potential impacts of russo-ukraine conflict and its psychological consequences among ukrainian adults: the post-covid-19 era. *Frontiers*, 2022.

[64] United Nations. Combined effects of war in ukraine, pandemic driving millions more into extreme poverty, senior united nations official tells second committee. *Press Release*, 2022.