Emotion Detection in Music and Emotion-Based Music Recommendation System

Aditi Roy
University of Illinois Urbana-Champaign
Champaign, United States
aditir5@illinois.edu

Kinjal Chaudhari
University of Illinois at Urbana-Champaign
Champaign, United States
kchaud9@illinois.edu

Linnea Marsh
University of Illinois at Urbana-Champaign
Champaign, United States
linneam2@illinois.edu

Manasi Mangalvedhe
University of Illinois at Urbana-Champaign
Champaign, United States
manasim2@illinois.edu

ABSTRACT

This project aims to develop a system that detects emotions in music by analyzing song lyrics and recommends songs to users based on their emotional preferences. By leveraging natural language processing (NLP) techniques for lyrics, the system will classify songs into emotional categories such as joy, sadness, and anger. The emotion detection component will be integrated into a recommendation engine that suggests music aligned with the user's emotional state. This project combines emotion recognition with machine learning-based recommendation systems to enhance personalized music experiences.

1 INTRODUCTION

Emotion plays a significant role in the way individuals interact with music. People often select songs that resonate with their current mood or seek music to evoke a desired emotional state. Existing music recommendation systems primarily rely on user interactions, genres, and historical data but often lack a deeper understanding of the emotional content within the songs themselves. This project proposes a novel approach by incorporating emotion detection in lyrics to create a more emotionally intelligent recommendation system. By analyzing the emotional tones of music, we aim to provide users with personalized recommendations that align with their emotional needs.

2 DESCRIPTION

The project will be divided into three main components: Data Collection and Preprocessing, Emotion Detection, and Recommendation System.

3 DATA COLLECTION AND PREPROCESSING

3.1 Data Collection:

The song lyrics we used were collected from the public Genius Lyrics dataset [1]. This is a credible source for music since Genius is a large organization and Genius's community of contributors powers the world's biggest collection of song lyrics and musical knowledge. It also includes a very large selection of music, which will ensure a diverse and representative sample of lyrical content across various genres, artists, and time periods, making our project more accessible and appealing to a wider audience.

We also used the NRC Emotion Lexicon [2], which provides a comprehensive list of words associated with various emotions. The lexicon categorizes words into a set of basic emotions, including joy, sadness, anger, fear, trust, surprise, anticipation, and disgust, as well as positive and negative sentiments. The lexicon is grounded in psychological research, making it a scientifically validated tool for emotion detection.

3.2 Preprocessing:

To make the dataset more robust, we first dropped all rows with null values. We also filtered the dataset to only include English songs since our lexicon is limited to English. Then, the data was cleaned by doing the following:

- **Tokenization:** The input is split into individual words.
- **Punctuation Removal:** Punctuation marks and special characters are removed (’",<>./?@$ % * /!()-[];:).
- **Lowercasing:** All tokens are converted to lowercase to ensure uniformity.
- **Stopword Removal:** Common stopwords (e.g., “and,” “the”) are filtered out using the Natural Language Toolkit (NLTK).
- **Lyrical Marker Removal:** Lyrical markers (e.g., [Chorus], [Verse]) are removed as they are not part of the lyrics, in order to ensure they don't affect the emotion distribution.

We also had to preprocess the lexicon in order to be able to effectively use it.

4 EMOTION CLASSIFICATION

4.1 Model Development:

To classify each song, we began with applying TF-IDF (Term Frequency-Inverse Document Frequency) weighting to the song lyrics to quantify the importance of each word relative to the entire dataset. The TF-IDF scores for the lyrics are computed, and these scores are then used in conjunction with the emotion labels from the NRC Emotion Lexicon to generate an emotion distribution for each song. Each song is analyzed to determine its emotion and sentiment distribution based on the weighted contributions of words from the lexicon.

CS 410, December 9, 2024, University of Illinois Urbana Champaign
.

- **Lexicon Matching:** Each token is checked against the NRC Emotion Lexicon. For each matching word, the corresponding emotions are counted.
- **Emotion Distribution Calculation:** The raw emotion counts are normalized to create a probability distribution over the emotion categories, ensuring that the sum of all emotion probabilities equals 1.

The output is an emotion distribution vector, representing the lyrics's emotional state as inferred from its lyrics.

The output of this process is a DataFrame containing:

- **song_id:** A unique identifier for the song.
- **artist:** The name of the artist.
- **title:** The title of the song.
- **dominant_emotion:** The primary emotion detected in the song.
- **dominant_sentiment:** The overall sentiment (positive or negative).
- **emotion_distribution:** A dictionary detailing the distribution of various emotions within the lyrics.
- **sentiment_distribution:** A dictionary detailing the distribution of sentiments (positive or negative).

4.2 Model Evaluation

In order to evaluate the emotion classification, we decided to perform a survey. Since there is limited data available on sentiment analysis from song lyrics, we used our survey results to create a ground truth to compare to our output. We selected 10 random songs from the dataset and had 10 people rate each song on how they felt the lyrics related to the emotions. Specifically, for each song they assigned numbers to each of the emotions we are considering: joy, sadness, anger, fear, trust, surprise, anticipation, and disgust. We then aggregated the results across participants and normalized the results.

To assess the performance of our emotion classification model, we implemented two statistical measures: Kullback-Leibler (KL) Divergence and Cosine Similarity.

4.2.1 Kullback-Leibler Divergence. The KL Divergence measures how one probability distribution diverges from a second, expected probability distribution. In our case, we compared the empirical emotional distributions derived from the survey with the predicted distributions generated by our emotion detection model. The survey-based distributions were created by aggregating and normalizing participant scores for each emotion, while the model-generated distributions were produced by analyzing lyrics using the NRC Emotion Lexicon and normalizing the resulting emotion counts.

The calculation takes these two empirical distributions as input, adds a small value to prevent division by zero or taking the logarithm of zero, and computes the sum of the divergence. Importantly, no theoretical distribution fitting (e.g., to a normal or uniform distribution) was performed. Instead, the KL Divergence directly compared the normalized relative frequencies from both sources.

We then looped through each song in the survey, comparing the true emotional distribution with the predicted distribution. A lower KL Divergence indicates that the predicted emotional distribution closely aligns with the participants' assessments, suggesting better model performance.

Table 1: KL Divergence Results

song_id	KL_Divergence
1062758.0	0.946462
4191823.0	0.155166
4313071.0	0.422628
5955393.0	0.292585
6047243.0	0.492483
6184434.0	0.228473
6636486.0	0.188995
6736901.0	0.379533
6850734.0	0.936562
7402191.0	0.327137

Considering that the scale of the divergence is $[0, \infty]$, our values are low, indicating that the predicted emotional distributions are closely aligned with the true emotional distributions rated by participants in the survey. Specifically, the KL Divergence values range from approximately 0.15 to 0.94, with the lowest values suggesting that the model effectively captures the emotional content of the lyrics for certain songs.

4.2.2 Cosine Similarity. In addition to KL Divergence, we calculated Cosine Similarity, which measures the cosine of the angle between two non-zero vectors. This measure helps assess how similar the predicted emotional distributions are to the ground truth provided by the survey responses. The cosine similarity values range from -1 to 1, where 1 indicates identical distributions and 0 indicates orthogonality.

We calculated the similarity matrix between the predicted emotional features and the survey responses. The resulting similarity scores were stored in a DataFrame, shown below, which illustrates how closely aligned our model's predictions are to the human-rated emotions.

Table 2: Cosine Similarity Results

song_id	self_similarity
1062758	0.126628
4191823	0.969375
4313071	0.986858
5955393	0.963733
6047243	0.996586
6184434	0.998435
6636486	0.054223
6736901	0.999222
6850734	0.998101
7402191	0.997302

The results reveal a strong alignment between our model's predictions and the human-rated emotional responses for most songs. Notably, some songs achieved cosine similarity scores above 0.99, indicating an almost perfect correspondence between the predicted and true emotional distributions. This suggests that our model successfully captures the emotional nuances of the lyrics for these songs.

Conversely, a few songs indicate that the model's predictions were not well aligned with the human assessments. This discrepancy highlights potential areas for improvement in our model, such as examining the lyrical content or adjusting the emotion detection parameters for this specific case.

4.2.3 Insights. Overall, the KL-divergence and cosine similarity analysis provides additional confidence in the effectiveness of our emotion detection model, particularly in its ability to accurately classify the emotional content of lyrics.

5 RECOMMENDATION SYSTEM

5.1 Model Development

The recommendation system leverages the emotional content of song lyrics to suggest songs aligned with a user's emotional state. The process can be divided into three key stages: input processing, emotion vector creation, and song recommendation based on cosine similarity.

To gather user input, the system prompts the user with a simple question: "How are you feeling today?" The user response is then preprocessed, similarly to the manner in which the lyrics were preprocessed, to extract meaningful tokens.

To map the user's response to an emotional state, the system employs the NRC Emotion Lexicon, similarly to the process used with the lyrics previously, and forms a user-specific emotion distribution vector, representing the user's emotional state as inferred from their input.

Each song in the dataset is associated with a precomputed emotion distribution derived from its lyrics. The system compares the user's emotion vector with each song's emotion vector using cosine similarity, which measures the angular similarity between two vectors. Higher similarity scores indicate a closer match between the user's emotional state and the emotional content of the song. The songs are ranked based on their similarity scores, and the top 10 songs are presented to the user as recommendations. The system provides a ranked list of songs, each accompanied by its similarity score. An example output is shown below:

```
---------------------------------------------------
Top 10 Songs Based on Similarity:
1. Tony was an ex-con by The Coronas - Similarity: 0.9918
2. Booksmart Devil by Silversun Pickups - Similarity: 0.9895
3. Roadhouse by Helldorado - Similarity: 0.9811
...
10. Views by Official_JFX - Similarity: 0.9678
---------------------------------------------------
```

5.2 Model Evaluation

The evaluation of a recommendation system typically involves metrics such as precision, recall, or user satisfaction surveys. However, given the unique nature of this system, which recommends songs based on the cosine similarity between a user's emotional query and the emotional distribution of song lyrics, traditional evaluation methods are less applicable.

In this system, recommendations are inherently based on maximizing the similarity between the user's emotion vector and the precomputed emotion vectors of songs. Thus, the similarity metric itself serves as both the recommendation criterion and the evaluation metric. Specifically, we compute cosine similarity using the following formula:

$$\text{Cosine Similarity} = \frac{\sum_{i=1}^{n} A_i \cdot B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \cdot \sqrt{\sum_{i=1}^{n} B_i^2}} \tag{1}$$

where A represents the user's emotion vector and B represents a song's emotion vector. A higher cosine similarity score indicates a closer match between the emotional content of the query and the song.

Since the system's primary goal is to recommend songs with high emotional congruence, we evaluated it by examining the consistency and robustness of the recommendations. The system consistently ranks songs with higher emotional similarity to the user's input at the top of the recommendation list. The cosine similarity score, used both for ranking and evaluation, ensures that the system selects songs that best match the user's emotional state. By design, the system's recommendation mechanism is directly tied to the evaluation metric. The recommendation process maximizes the cosine similarity score, making the system inherently robust. As such, the evaluation aligns with the system's objective function.

In summary, the current evaluation framework, based on cosine similarity, is a robust and logical approach given the system's design. The recommendations are both generated and evaluated based on their alignment with the user's emotional input, ensuring that the system fulfills its intended purpose of providing emotionally resonant music suggestions.

6 INSIGHTS

6.1 Model Performance and Improvement Areas

The songs with lower similarity scores suggest that the model may struggle with certain types of lyrical content, such as abstract or metaphorical language. This could stem from the limitations of the NRC Emotion Lexicon, which might not cover all relevant words or contextual meanings. Addressing these issues could involve:

- **Expanding the Lexicon:** Incorporating additional emotion lexicons or context-aware embeddings like BERT or GPT-based models to better handle nuanced language.
- **Contextual Emotion Detection:** Using contextualized models to understand the sentiment of phrases rather than individual words, allowing for more accurate emotion mapping.
- **Weighted Word Contributions:** Assigning weights based on the position of words in lyrics (e.g., chorus vs. verses) to capture the emotional emphasis.

6.2 Insights for the Recommendation System

The recommendation system's reliance on cosine similarity between user input and song emotion vectors has proven effective in aligning song suggestions with emotional needs. The top-ranked songs consistently exhibit high similarity, reflecting the model's ability to personalize recommendations. However, several areas warrant further exploration:

- **Handling Ambiguous Inputs:** Enhancing the system's ability to interpret vague or multi-faceted user inputs by generating composite emotion vectors.
- **Diversity in Recommendations:** Introducing mechanisms to ensure a diverse range of recommendations across genres, artists, and moods while maintaining emotional congruence.
- **User Feedback Integration:** Incorporating user feedback loops to refine recommendations over time, adapting to individual preferences and improving personalization.

6.3 Future Directions and Evaluation

Building on the current framework, future work will focus on developing a content-based filtering system inspired by Word2Vec. This involves representing both user input and songs as high-dimensional emotion vectors, enabling more sophisticated similarity calculations and clustering.

6.3.1 Proposed Enhancements.

- **Vector Space Models:** Representing songs and user inputs in a shared vector space, where proximity reflects emotional alignment. This could facilitate not only recommendations but also playlist generation.
- **Dynamic User Profiling:** Allowing users to save profiles based on their emotional preferences, enabling the system to provide context-aware recommendations over time.
- **Real-time Adaptation:** Implementing real-time updates to song rankings based on shifts in user mood or context, enhancing responsiveness.

6.3.2 Future Evaluation Considerations. While cosine similarity serves as a useful metric, broader evaluation methods will strengthen the system's credibility and user satisfaction:

- **User Studies:** Conduct surveys and usability tests to gather qualitative feedback on recommendation relevance and emotional resonance.
- **Diversity and Novelty Metrics:** Introduce metrics that evaluate the diversity and novelty of recommendations, balancing emotional alignment with variety.
- **Comparative Benchmarking:** Compare the system's performance with existing emotion-based or conventional music recommendation systems to contextualize its effectiveness.

REFERENCES

[1] Carlos G. 2023. Genius Song Lyrics with Language Information. https://www.kaggle.com/datasets/carlosgdcj/genius-song-lyrics-with-language-information Accessed: 2024-12-04.

[2] Saif M. Mohammad and Peter D. Turney. 2013. Crowdsourcing a Word-Emotion Association Lexicon. In *Computational Intelligence*, Vol. 29. Wiley Online Library, 436–465. https://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm