Automating Spotify Playlist Creation Based On Emotion

Bella Baidak (bb568), Christine Shen (cs952)
CS 5785: Applied Machine Learning
Cornell Tech
New York, NY

January 3, 2022

Abstract

Over the years, there has been a push for machines to become more emotionally aware. Music elicits varying degrees of emotions in humans, so in this project, we explore our emotional connection with machines by using machine learning to generate happy and sad playlists. To do so, we curated two song datasets from a kaggle dataset, in which we manually labeled each song with an emotion, and a Spotify dataset with songs pulled directly from existing happy and sad playlists. In our experimentation, we used the bag of words model and trained multiple logistic regression classifiers. At first, we tried just using song lyrics in our models, but after getting emotionally confusing predictions, we decided to also incorporate audio features. Despite the model trained on the manually-labeled dataset performing the best based on F1 score, we found that models that included audio features produced the best results.

1 Introduction

Currently, Spotify has musical experts manually creating a plethora of playlists on its platform, with some organized by genre and others by mood. We are trying to see if we can automate this process and create more user-curated content by allowing for playlist generation based on human emotions and song lyrics. We decided to focus on happy and sad emotions as these emotions are most distinguishable in music. Other basic human emotions like disgust and surprise are more subjective across people and are not typically used to categorize music.

This work would contribute to the field of affective computing by enabling a system to gain an understanding about human emotions based on language. Music can be very emotionally evocative for humans, so creating a system that is able to recognize emotions through music could help strengthen our relationship with machines.

2 Background

Our project draws on two important and emerging fields in computer science, affective computing and sentiment analysis. Affective computing is the study and development of systems and devices that can recognize, interpret, process, and simulate human affects [1]. Affective computing technologies sense the emotional state of a user through sensors, microphone, cameras, language, audio, and/or software logic. Sentiment analysis is similar to affective computing but is specific to language. It has become a popular natural language processing technique used to identify and extract sentiments, attitudes, opinions, and emotions from text towards a topic, person, or entity.

Supervised learning algorithms have been commonly used for these techniques, such as Naive Bayes, Support Vector Machines, and linear classifiers like Logistic Regression. More recently, deep neural networks and unsupervised learning and hybrid approaches have become increasingly prevalent in performing these tasks.

To appropriately prepare data for classification, various preprocessing techniques need to be applied. One such important method is the Bag of Words model, which is a common approach used to simplify text representation. Text is reduced to a multiset (bag) of words in which the (frequency of) word occurrences is measured. This is particularly useful when word frequency or occurrence is a key feature needed for a classification task.

Music Emotion Recognition is a task relevant to both sentiment analysis and affective computing. There has been some previous work that have attempted to classify music by emotion. Most recently, a study used sonification and machine learning to predict emotions from audio [2]. Though this study found interesting and relatively accurate predictions, they did not consider lyrics when doing this research, which we believe is an important part of making a song emotional. A different study similar to our project used lyrics and naive bayes classifier to make emotional predictions in songs [3]. While this study did have high accuracy in prediction of happy songs, there was no mention of validating the emotional feeling of the predictions by listening to the predicted songs.

Another interesting application of Music Emotion Recognition is the exploration of a mood-based music recommendation system for car drivers, which aims to enhance their driving experience, comfort, and cautiousness [4]. This study attempts to apply sentiment analysis to something we interact with in our day-to-day lives, highlighting an example of how emotional awareness in systems can be enhanced through music.

With the amount of data surrounding the music we listen to, prominent music apps, such as Spotify, are already applying some machine learning techniques to this data, such as personalized song recommendations. Yet, there are still many untapped opportunities in such apps that are still manually curated, one in which we explore here.

Spotify collects many different features for each song, which includes not only the artist and album it is associated to, but also lyrics and various audio metrics, such as danceability, tempo, and energy. With this data available, it is feasible to extend the application of sentiment analysis and affective computing into this space.

3 Method

We used common Natural Language Processing techniques that we learned in class. We are specifically using the Bag of Words model and logistic regression, a classification model that uses the logistic function to predict the probability of belonging to a certain class or label. To account for overfitting, we used L2 regularization, which penalizes large learned parameter values by increasing the error (cost) function. We chose to use Logistic Regression over an alternate method such as Naive Bayes because we found in our past experiments in the course homeworks, Logistic Regression produced more accurate results.

Since music is subjective and there isn't an existing dataset that contains emotion-labeled songs, we decided to take two approaches in creating the labeled dataset ourselves.

3.1 Approach 1

In our first approach, we wrote an algorithm to individually label each song from a Spotify kaggle dataset of 18,000 songs [5]. This dataset contains many song attributes, including the song artist, album name, song lyrics, and even tempo and danceability. Also, the dataset contained non-english songs, so we filtered those out. We first found a NRC Word-Emotion Association Lexicon dataset [6] containing English words and their associations with 8 basic emotions. Focusing on just the two core emotions we planned to target, happy and sad, we wrote an algorithm that aggregated all words associated with each emotion into their respective lists. Then, for each song, we counted the number of occurrences of each word appearing in the lyrics and labeled the song with the emotion that had the maximum number of associated words appearing in the song. We added an isHappy column labeled 1 if a song is happy or 0 otherwise for training and testing purposes.

3.2 Approach 2

Our second approach was to create a dataset from existing playlists on Spotify categorized as happy or sad. To do this we used Spotify's API to pull the song list and information from the Spotify playlists and create a dataframe in python. Once we had a dataframe containing both happy and sad song information, we then used Genius API to get the song lyrics for each song and added them as a column in the dataframe. We removed all duplicate songs, songs whose lyrics could not be found on Genius, and songs that had lyrics not in English from the dataframe. Lastly, we added a column isHappy and marked the happy songs with a 1 and the sad songs with a 0. In total, there were 5568 songs.

3.3 Preprocessing and Training

Once we had our datasets, we used standard preprocessing methods on all of the song lyrics. First, we set all words to lowercase. Using the nltk package, we then used the WordNetLemmatizer to lemmatize all words and strip punctuation and stop words. We then built a vocabulary of words from a subset of each dataset for training. We represented each song with the Bag of Words model, using the CountVectorizer from sklearn with the binary parameter set to True and minimum occurrence threshold (min_df) of 5. For each emotion, we proceeded to train a Logistic Regression classifier with l2 penalty on the vectorized lyrics. We evaluated the performance of our model by performing error analysis with the F1 score between the songs our model labeled as, say, happy and the 'true' (happy) songs. Finally, we compared the performance of each model with one another.

While we mainly focused on song lyrics, we did some experimentation in combination with some of the other available song audio features. After we created several variations of models using the methods described above, we wanted to validate our results by creating actual Spotify playlists. To do so, we used the 'Create Playlist' API from Spotify and randomly selected 100 songs to include in each playlist.

4 Experimental analysis

Given the experimental nature of our project, we tried several different models with different variations of our datasets to see what would achieve the best performance. At first, we created two bag of words models using a subset of 1000 songs lyrics from the labeled data set we created. After training one model on songs labeled as happy and one model on songs labeled as sad, we tested our models on a subset of 1000 songs from the original kaggle dataset. We then calculated the F1 scores for both models. The F1 scores of the models trained on 1000 songs were very low, both around 30 percent accuracy. Because of this low accuracy, we began increasing our sample size of songs by 1000. We found that the F1 scores reached peak performance of about 70 percent accuracy with 6000 songs.

Next, we created a happy and sad model from the Spotify datasets. These models contained the full Spotify dataset that was just under 6000 songs. After training these models, we tested them on a subset of 6000 songs from the kaggle dataset. These models yielded slightly lower F1 scores of around 65 percent accuracy.

Next, we were curious to see if combining a subset of the labeled kaggle dataset and Spotify dataset would increase performance of our models. We thought this would combine the human touch of the Spotify dataset and the verbal logic of the labeled data. So, we trained another two models (one happy and one sad), each containing a random subset of 3000 songs from each dataset (6000 songs in total). We then trained these models on a subset of 6000 songs from the kaggle dataset. The accuracy of these models were very similar to the other models and were just under 65 percent accurate.

We then thought it would be interesting to actually create playlists from our results because the accuracy of the models here was less important then the emotional feeling of the predictions. The playlists created from all six models were rather emotionally confusing, containing songs that were not clearly happy or sad. This finding led us to experiment further.

Considering using just lyrics alone gave us emotionally confusing results, we decided to experiment other metrics to determine the emotion of a song. In the datasets found on kaggle and how the data comes in from Spotify, there are several audio features included with each song. To figure out what audio features might be important to train on, we grouped the data by happy- and sad-labeled songs and took the average of all the audio features. When comparing the average value of each audio feature for happy and sad songs, we noticed a considerable difference in values for 'danceability', 'energy', 'loudness', and 'valence' audio features in the Spotify data set. For example, the average valence of sad songs was 0.340647 and the average valance of happy songs was 0.628054. Interestingly, we did not observe the same trend with the labeled kaggle dataset we curated.

These findings with audio features next led us to create two additional models. Both models used the entire Spotify data set for training. The first model simply included the values 'danceability', 'energy', 'loudness', and 'valence' audio features. The second model included the same audio features and the song lyrics. We tested both models on a subset of 6000 songs from the kaggle dataset. Given the trend in audio features was not observed in the labeled kaggle dataset, there was no reliable source of truth to compare these models to. Thus, we did not compare any F1 scores. Instead, we decided to again create playlists from these predictions and listen to the playlist to test their performance. Unfortunately, adding audio features only slightly improved the quality of the outputted playlists.

5 Discussion and Prior Work

Model Performance		
Model	F1-score (Happy)	F1-score (Sad)
Dataset 1	0.666	0.701
Dataset 2	0.65	0.65
Combined Datasets	0.634	0.644
Audio Features	-	-
Audio Features and Lyrics	-	-

The performance of each model is summarized in the above table. Links to the curated Spotify playlists can be found in the Appendix.

Out of all of our models, the Dataset 1 model performed the best with the highest F1 score. It appeared to be the closest to the source of truth in terms of lyric content, as it was labeled in such a manner based on the appearance of happy versus sad words.

However, there were mixed results when listening to the playlists. In the happy playlists, there was a mix of upbeat, lighthearted songs, often associated with joy, and slow, mellow songs, which often has sorrowful undertones. The same was observed for sad playlists. Yet, there were instances in each where upbeat songs were rooted in sadness and mellow songs were rooted in happiness. This dichotomy shows that lyrics alone cannot be used to determine whether a song is associated to one specific emotion.

Overall, the playlists generated from models that included audio features tended to be less confusing than the other playlists. In these playlists, there were no obvious instances of sad songs appearing on the happy playlist and vice versa. In contrast, the other playlists were generally all over the place. For example, the song 'Creep' by Radiohead appeared on a happy playlist generated from a lyrics-only model.

Prior research has focused on predicting the emotion in songs from song lyrics and audio separately, producing sub par results. Emotion is comprised of multiple modalities [7], so for a machine to truly understand emotions, these modalities must collectively be considered in predictions. Therefore, it makes sense that the playlists generated from both song lyrics and audio features were the least emotionally confusing. This demonstrates that multi-modal emotion recognition is the key to developing human connections to machines.

6 Conclusion

Overall, there was no clear method we used that achieved emotionally strong playlists for either happy or sad emotions. The key finding from our experiments is that using words alone is not an accurate way to measure emotion in songs. This draws from the ambiguous nature of classifying songs by emotions. A song I may perceive as happy may not also be perceived as happy by another person. Additionally, a song could have depressive or dark lyrics with an upbeat tempo or vice versa. For example, the song 'Pumped up Kicks' by Foster The People has a dark message, but is very catchy and upbeat.

If this project was to be continued, there are several things we could do to try and increase emotional accuracy in our generated playlists. First, more research could be done on the psychological aspects that make up emotion and how those modalities are present in music. If this was done, we could perhaps try to add more of these modalities into our models. The next thing that could be done is gain user feedback. Because emotion in music is very subjective, we could create an experimental interface where users could rate songs as happy or sad and then give this data back to a model to learn from. This would allow the model to gain direct emotional understanding from users. Lastly, we could try to assign weights to certain words in our bag of words model that are more likely to indicate happiness or sadness.

This project leaves us with questions of: Is it possible to create a universal set of happy and sad music? Or, is happy and sad music generated by machine learning more achievable on a personal level? As the field of affective computing continues to grow and we continue to become more connected with machines, perhaps these questions will be answered.

References

[1] Picard, R. W. (1995). Affective Computing. M.I.T Media Laboratory Perceptual Computing Section Technical Report No. 321.

[2] Abri, F., Gutiérrez, L. F., Datta, P., Sears, D. R., Siami Namin, A., amp; Jones, K. S. (2021). A comparative analysis of modeling and predicting perceived and induced emotions in sonification. Electronics, 10(20), 2519. https://doi.org/10.3390/electronics10202519

[3] Raschka, Sebastian. (2016). MusicMood: Predicting the mood of music from song lyrics using machine learning.

[4] Çano, Erion et al. (2018) Text-based Sentiment Analysis and Music Emotion Recognition. Polytechnic University of Turin.

[5] Nakhaee, Muhammad. (2020) Audio Features and Lyrics of Spotify Songs. https://www.kaggle.com/imuhammad/audio-features-and-lyrics-of-spotify-songs.

[6] Saif, Mohammad. (2021) NRC Word-Emotion Association Lexicon. https://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm.

[7] Zhao, Sicheng et al. "Emotion Recognition From Multiple Modalities: Fundamentals and Methodologies." IEEE Signal Processing Magazine 38.6 (2021): 59–73. Crossref. Web.

A Appendix

Curated Playlists

- Dataset 1
  - Happy:
    https://open.spotify.com/playlist/4a59BYLVwShht3AQGfy5XA?si=6239971ba8184bf6
  - Sad:
    https://open.spotify.com/playlist/5PacWMqLKeYAcv1s4Q4WCT?si=dd73b07d750247cd
- Dataset 2
  - Happy:
    https://open.spotify.com/playlist/13WYgcPLIC7EKGVFareF6z?si=3d683549c8ed4354
  - Sad:
    https://open.spotify.com/playlist/7Efih4u6rw2qdjgWIuG7NZ?si=6f1f456ea9204bea
- Combined Datasets
  - Happy:
    https://open.spotify.com/playlist/2BoKXDL7L1i8yYtk38iYVP?si=e7018758d4f94b73
  - Sad:
    https://open.spotify.com/playlist/2VCj4LT5SOcsa5WdHBfA70?si=d566c2ef441b4e1b
- Audio Features
  - Happy:
    https://open.spotify.com/playlist/5u5KHlhirlVSznJUC0Ta9H?si=2beb65fdf0ab4bd3
  - Sad:
    https://open.spotify.com/playlist/2A03qMx2HtO6TGxZJQcf2N?si=08e67e07293e4c44
- Audio Features and Lyrics
  - Happy:
    https://open.spotify.com/playlist/2GyBV9ciZXFlSCYfNB0NSP?si=d681e75d9115438b
  - Sad:
    https://open.spotify.com/playlist/2iTrmmaiJPwUN0VmpmFwT8?si=854927d2db364077