EECS 4404/5327 Project Part 5 Draft Report

Abstract

Our application is about classifying songs by mood and using that classification to generate playlists with a certain mood and artists. The user will input their liked songs or a playlist of songs from Spotify and using a neural network we will classify each song by its mood. The app will then use these mood classifications to generate a playlist based on what mood and artists the user chooses to select. This allows users to generate specified sub playlists on the go without going through the effort of curating a playlist themselves.

Introduction

1. What is your application?

Classification of music is an important task that allows users to specify what kind of music they want to listen to in the moment. Generally, music is identified based off genre, however this does not fully encompass what mood of music you desire to listen to at certain moments. To address this problem our application classifies music into mood labels: Energetic, Happy, Sad, Calm.

2. What are the assumptions/scope of your project?

Scope is limited to Spotify users in terms of importing user playlists/liked songs to categorize into mood subsets, non-Spotify users will not be able to use this tool. The songs chosen for this playlist are limited to the songs the user inputs.

Assumptions:

- music/song is available on Spotify.
- user has a Spotify account with liked playlists/songs

3. Justify why is your application important?

Spotify has 489 million monthly active listeners as of 2023. All these users can utilize this application to create a playlist for certain moods combined with criteria they specify.

Many of these users do not put in the effort into curating their playlists, either having very large and scattered playlists theme-wise, or they simply have no playlists and shuffle through their liked songs. Being able to automatically curate these lists into smaller subsets will help users find the music they want at the time they want it.

4. Similar applications

The “Towards Data Science” article [3] -- on predicting the mood of a song with deep learning -- implements a similar application of music mood classification. Our application is similar to this but utilizes a larger and more updated dataset.

Spotify mixes: Takes your liked artists and curates playlists based on categories such as mood or genre. E.g., “Chill Mix” or “Hip Hop Mix”

MOOD website [6] which creates a playlist based on mood and artists selected while adding recommended songs based on personal listening history.

Our application would be different in its ability to specify which songs to include in the pool when creating the playlist, as these playlist generators will add songs from other artists that you don’t listen to, or songs you don’t like from an artist that you do like. Also, you will be able to specify certain artists along with a mood.

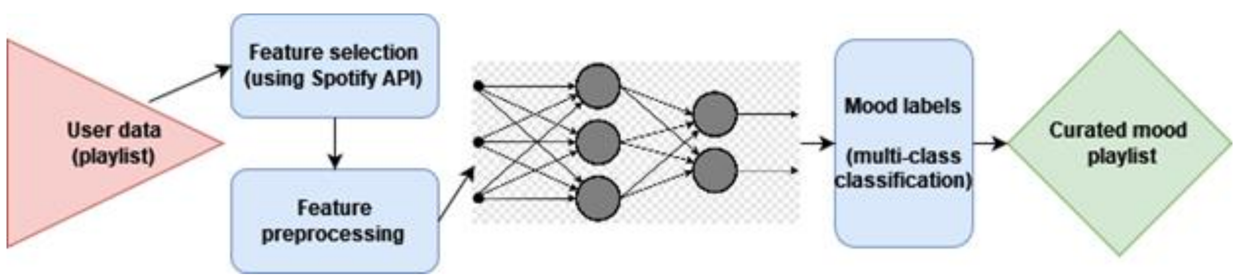
These improvements are important because it allows you to build playlists that are highly specific to you in the spur of the moment without having to actually curate and maintain them yourself.

5. Adjustments to part 1

The dataset used is now extracted from Spotify and is stored in a csv file where each line represents a song and info associated with that song separated by commas. In part 1 we mentioned analyzing the audio itself to extract features, we are no longer doing so, currently we depend on Spotify for the audio analysis by using spotipy library.

Methodology

1. Design/pipeline



The design pipeline starts with user data. The input is a Spotify playlist which all song IDs are extracted from. Next in the pre-processing stage, the features of each song are gathered using the Spotify API's audio analysis. Next the features of each song are fed into the neural network, to get a mood prediction. The neural network consists of an input layer of 10 nodes and two layers. The first hidden layer has 8 nodes and the activation function chosen was ReLU. The second and final layer has 4 nodes with activation function softmax.

Once the prediction for each song in the playlist is computed, the post-processor groups the songs by mood and genre, and subsequently produces a playlist for each mood subset of the original playlist.

2. Dataset

The dataset consists of 1044 songs scraped from the spotify API and labeled by mood. Initially, we planned to curate a labeled dataset from scratch, but due to time constraints we used a 687-song dataset pre-labeled by GitHub user cristobalvch [5], later extending the dataset to 1044 songs via group consensus, analyzing music and selecting a label with the methodology outlined in [2] in mind. The data is stored in a csv file where each of the 1044 rows is represented by:

- Name, album, artist, id, release_date, popularity (auxiliary data provided by the Spotify API, not included in the feature set)
- length, danceability, acousticness, energy, instrumentalness, liveness, valence, loudness, speechiness, tempo, key, time_signature (selected for inclusion in the feature set)
- Mood (label)

Given the format of the features as-is upon being scraped from the Spotify API, no further data pre-processing was required. For example, we observed that the chosen features seem to already be well normalized.

3. Model training

For our neural network we used 10 features of the song, generated by calling the Spotify API, that include length, danceability, acousticness, energy, instrumentalness, liveness, valence, loudness, speechiness, and tempo. We decided it would be easier to have the model be multi-class classification as opposed to being multi-label in order to have a more intuitive output. The model then outputs one of four moods: energetic, sad, happy, or calm.

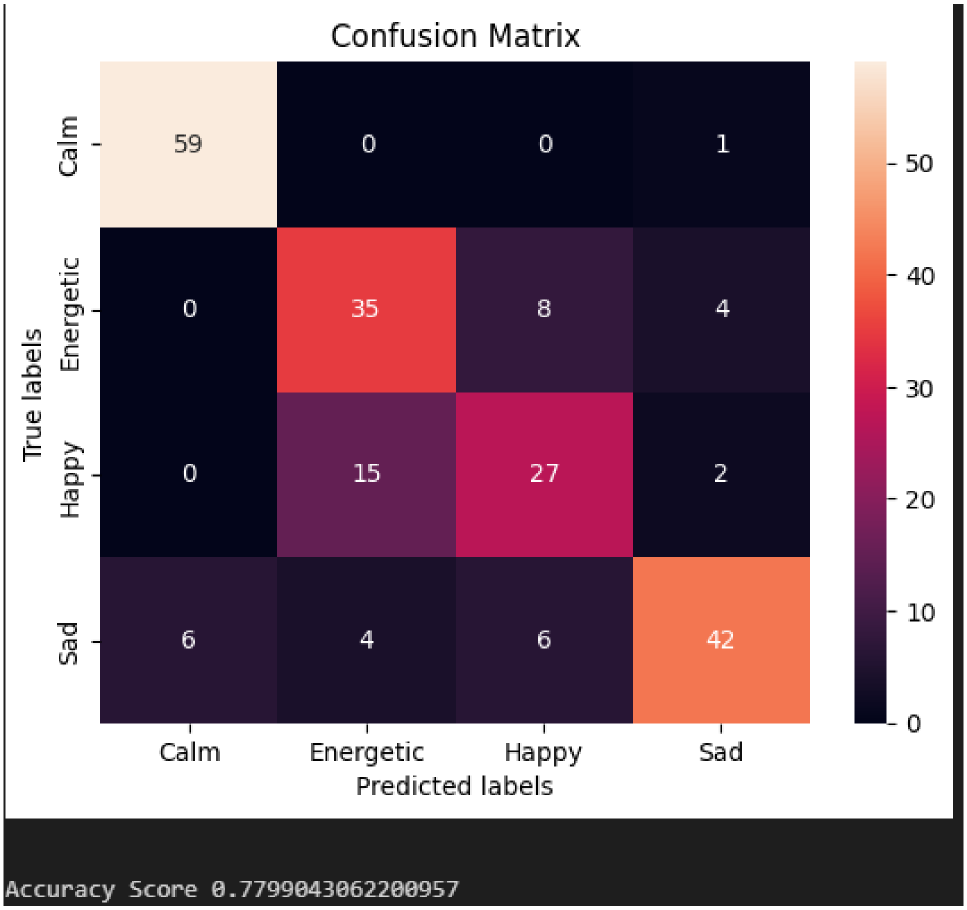
The studies in part 2 weren't applicable to our project given the circumstances.

4. Prediction

The model predicts the output by accessing the features of the song from Spotify via the API and feeding it into the multi-class mood classifier model, outputting one of the four previously outlined mood labels.

Results

1. Evaluation



Firstly, we test the model by k-fold cross-validation to obtain a baseline predication accuracy for the model. To gain more localized insight into the model's performance, we also create a confusion matrix comparing the pairwise predicted and true label results. The most significant local point of inaccuracy appears to occur in classifying examples where the true label is Happy; the model was in fact more likely to classify those examples as Energetic. The confusion matrix can be seen below. Final update: After extending the dataset, a significant improvement in the model's accuracy in correctly classifying songs as Happy was observed.

2. Results

To assess the accuracy of our model, we chose to employ k-fold cross-validation on the training set choosing k=10. Our reasoning behind choosing this method was that we considered that our labeled dataset may have been too small to split into a discrete training and test set without negatively impacting the quality of our model.

The mean accuracy of the k-fold cross-validation test sets was 77.96%, with a standard deviation of 3.49%.

Discussions

1. Implications

We believe the accuracy metrics outlined in our evaluation to be acceptable for our application, in terms of numbers.

Though we have not been able to conduct a formal user study as of now, not even the members of our group could always come to a consensus on the prevailing mood of a song in our discussions. We believe it is it reasonable to extrapolate that the personal and subjective nature of music mood analysis will create a significant degree of noise in any human consensus. In light of this, we believe blindly aiming for a higher baseline accuracy may not necessarily be the best approach to improving the usefulness of the model.

However, this also means we cannot conclude that the model is effective simply based on its accuracy, and it would be prudent to conduct user studies and consider domain expert knowledge.

2. Strengths

Our application takes full advantage of the data provided by the Spotify API; anybody who uses Spotify (or provides a playlist from a third party if they don't personally use it) can make use of our application to classify their playlist by a desired mood with nothing more than the playlist ID.

Consequently, there is a low barrier to our application being able to effectively serve our target user base. We believe it is safe to say that given a user-friendly UI, almost anyone could make use of our machine learning model to enrich their interactions with the music they love in just a few clicks.

3. Limitations

The music that is being classified by our project must be available on Spotify. This limitation is mitigated by the size and scope of Spotify's music library (over 100 million tracks).
We depend on Spotify's analysis of audio qualities, so we must rely on Spotify to be consistent in their audio analysis. Otherwise, the validity of our model would be dubious since we select our features entirely from the Spotify API's provided metrics.

4. Future directions

As for future directions, we would seek to extend the scope and nuance of our mood classification model. As the scope is currently limited to classifying a set of songs into one of 4 moods, we believe the system would benefit from diving deeper into domain knowledge in music and emotions and considering incorporating a larger range of mood labels in the model. We believe it would also be worth testing a multi-label implementation.

In order to best serve a potential user, we would also like to create a webapp interface for our model that allows a user to create playlists and gives song recommendations based off mood and other specifications.

Additional Questions

1. What are the feedbacks that you found useful from the peer evaluation?

“As you mentioned that song classes are somewhat subjective, and you are currently having some confusion between classes. I think that this could be turned into a feature where you can take the user bias on what they believe this song would fit in. This tailors the application more to the individual user.”

“For the idea, it would be helpful to consider how the mood classification could be refined to better match user preferences”

“For the neural network, considering the inclusion of additional features, such as lyrics or genre, could potentially improve the accuracy of the mood classifications.”

2. What changes did you make based on the feedback from peer evaluation?

We would like to make changes based on the user’s subjective opinion about the music, however that isn’t possible without retraining the model. I agree that including additional features like lyrics or genre could help in improving the accuracy of the model. An overall sentiment analysis can be done on the lyrics and that could help in determining the difference between happy and energetic, however that seems too difficult to implement into our project within our time constraints, and quite a lot of work for one feature. Adding genre would also help in contextualizing what certain moods would look like for each genre, but without a way to quantify a genre, we could not use it as a feature in our neural network besides one-hot encoding genres. Maybe an expansion of the pipeline could be considered, one that would take the current mood likelihoods and interpret the results based on genre somehow, however this could slow down performance.

References

1. https://www.statista.com/statistics/367739/spotify-global-mau/ (Spotify monthly users)
2. https://sites.tufts.edu/eeseniordesignhandbook/2015/music-mood-classification/
3. https://towardsdatascience.com/predicting-the-music-mood-of-a-song-with-deep-learning-c3ac2b45229e
4. https://github.com/cristobalvch/Spotify-Machine-Learning/
5. https://github.com/cristobalvch/Spotify-Machine-Learning/blob/master/data/data_moods.csv
6. https://www.moodplayl.ist/mood-playlist-generator-personalised-taste/