# Spotify Pop Music Analytics and Context-based Recommendation System

**Ruolin Hao**
University of Michigan
haorlin@umich.edu

**Mu Ding**
University of Michigan
dingmu@umich.edu

**Zhiang Zhang**
University of Michigan
zhiangz@umich.edu

**Abstract**

Over the past ten years, there has been a significant shift in the way people listen to music due to the rise of music streaming services. These platforms offer access to a vast number of songs, but with so many choices available, it can be challenging for users to find music they enjoy. To address this issue, music recommendation systems have become increasingly popular, as they aim to suggest music that matches the user's preferences based on their current situation. This project aims to analyze the characteristics of pop music and build a context-based recommendation system based on user preferences with Spotify data. By effectively utilizing the interaction effects between user context and track information, this application will provide users with a personalized pop music playlist and enhance user experience when enjoying music.

## 1 Introduction

Nowadays, music has become an essential part of our life, especially with the rise of music applications like Spotify, Apple Music, Youtube music, etc., people can listen to millions of songs anytime and anywhere, however, the downside is that there are too many options for people to choose from. As a result, the music recommendation system occurs and creates personalized playlists for users based on their moods, music preferences, and interested singers.

In this report, we firstly conduct pop music analytics with Spotify to provide different features, where people can identify how music evolves based on different artists, and then create a personalized context-based playlist for users based on different contexts: energizing, relaxing, focusing and commuting. While working on this project, we are ensuring that the data collected is necessary for the project and that the privacy of individuals is protected since privacy issues are essential for a data science project. Users can gain insight about music trends and customize their music playlist from the project. It is essential to consider the music recommendation system may shape people's preferences for the music, therefore it potentially has an impact on the music culture.

## 2 Related Work

Nowadays, scholars have done various research about music recommendation systems and three main music recommendation system categories are collaborative filtering, content-based, and context-based recommendation systems.

The collaborative filtering method is based on a collection of users' behavior, activities, and preferences using explicit and implicit collection methods. The main machine learning to predict people's music playlist is k-nearest neighbors. However, the content-based method is more complicated than collaborative filtering, since it mainly focuses on the features' importance and how to connect those features to people's preferences. To determine features the popular machine learning algorithm applied is Naive Bayesian classification or cluster analysis (Srebrenik, 2021). The context-based method generates the recommendation based on the public perception of the song. Scholars can collect data from different kinds of social media and then use a similar method, which can be considered as user-based context music recommendation, and the other is environment context-based to provide environment factors such as location to provide music recommendations (Paul & Kundu, 2020).

Girsang et al. (2019) created a collaborative filtering recommendation system using the Pearson correlation function to get the users' taste similarity. Niyazov et al. (2021) conducted two approaches to building content-based recommendation systems, where the first method is to implement acoustic feature analysis and the other is to apply the deep learning method such as artificial neural networks (ANN) combining computer vision to develop the recommendation system. Han et al. (2010) proposed an emotion state transition model (ESTM) to simulate the changes with the transition of people's moods, evaluated people's moods based on different situations, and with the help of a beat tracking algorithm, filtering the important features, then finalized the music recommendation system.

Also, there are some studies related to the hybrid system to combine those recommendations. Omowonuola et al. (2022) combined collaborative filtering, content-based, and context-based to create recommendation systems with the help of convolution neural network (CNN) and weight extraction method. Kathavate (2021) designed a hybrid music recommendation system combining collaborative filtering and content-based filtering together, where storing different features using k-means algorithm and leveraging cosine similarity method to find similarity between different users.

## 3 Data

### 3.1 Data Collection

The data for this project is accessed via Spotify Web API with a lightweight Python Library - Spotipy. With Spotipy, we are able to get full access to all of the music data provided by the Spotify platform. We collected 24,000 instances of tracks released from 2000 to 2023. In the database, each track instance contains 24 columns, including general information about the artist name, track name, track id, popularity, release date, duration, analysis url, as well as a series of music features, such as danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence, tempo, etc. The concatenated general information and audio features for a sample of 5 track instances are shown in Table 1. The descriptions of audio features are documented in **Spotify for Developers.**

Table 1. Data View

| | artist_name | track_name | popularity | danceability | energy | key | loudness | mode | speechiness | acousticness | instrumentalness | liveness | valence | tempo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **456** | Justin Bieber | Peaches (feat. Daniel Caesar & Giveon) | 81 | 0.677 | 0.696 | 0 | -6.181 | 1 | 0.1190 | 0.3210 | 0.000000 | 0.420 | 0.464 | 90.030 |
| **983** | Coldplay | Viva La Vida | 0 | 0.486 | 0.617 | 5 | -7.115 | 0 | 0.0287 | 0.0954 | 0.000003 | 0.109 | 0.417 | 138.015 |
| **146** | Taylor Swift | cardigan | 81 | 0.613 | 0.581 | 0 | -8.588 | 0 | 0.0424 | 0.5370 | 0.000345 | 0.250 | 0.551 | 130.033 |
| **299** | Doja Cat | Kiss Me More (feat. SZA) | 78 | 0.764 | 0.705 | 8 | -3.463 | 1 | 0.0284 | 0.2590 | 0.000089 | 0.120 | 0.781 | 110.970 |
| **7** | Miguel | Sure Thing | 90 | 0.684 | 0.607 | 11 | -8.127 | 0 | 0.1000 | 0.0267 | 0.000307 | 0.191 | 0.498 | 81.001 |

### 3.2 Data Preprocessing

First, the duplicate track instances are dropped from the data, and 23,987 unique instances are stored in our dataset. The audio features of the tracks are fundamental to the construction of our recommendations. We investigated the data types, ranges, and representations of various features to define the input variables of our model. For example, mode and key are categorical features with integer types, while tempo has specified and absolute quantitative values so that its range is not the same as other numerical features.

Next, all the features are standardized by removing the mean and scaling to unit variance. According to a study conducted by Pichl and Zangerle on User models for multi-context-aware music recommendation, they investigated 8 acoustic features in Spotify playlists and aggregated them using the arithmetic mean. Since mean absolute deviation (MAD) is a robust measure with respect to outliers, they have found that except for loudness, the variance of each characteristic inside a playlist is low and the MAD is rarely higher than the mean except for loudness. Therefore, we believe that loudness is an inconsistent feature even within a playlist and not relatively informative for user context representation. Finally, the rest of the 7 features were selected as the recommender system input, including danceability, energy, speechiness, acousticness, instrumentalness, liveness, and valence.

Besides, Principal Component Analysis (PCA) is conducted to support our analysis. The results of the PCA reveal the principal components, which can be used to understand the variations between tracks and determine the total variance explained by each principal component, as shown in Fig. 1. The accumulated variance of the principal components is 88.31%. Based on the PCA analysis, we can observe that tracks with a significant influence from energetic and acoustic features are separated from other tracks by the first principal component (PC1). Additionally, we are also able to distinguish the tracks with high or low danceability characteristics by PC1 and PC2. When we set up the contextual categories for users and compute recommendations, these features are more prioritized in our consideration during labeling.

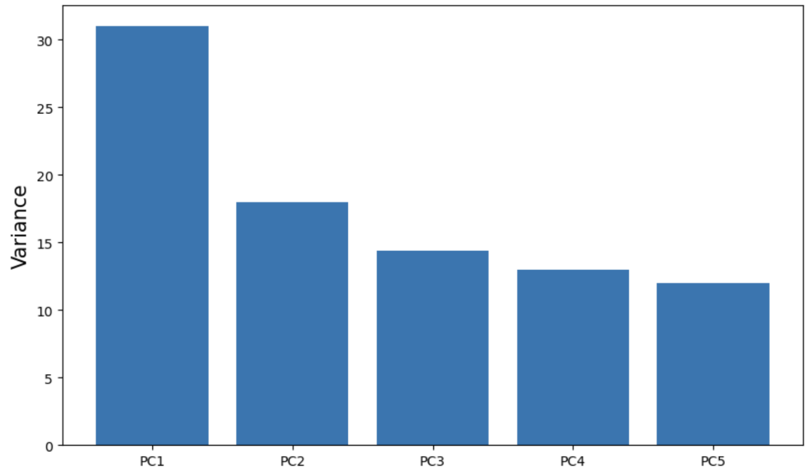



Fig. 1 Explained variance by each principal component

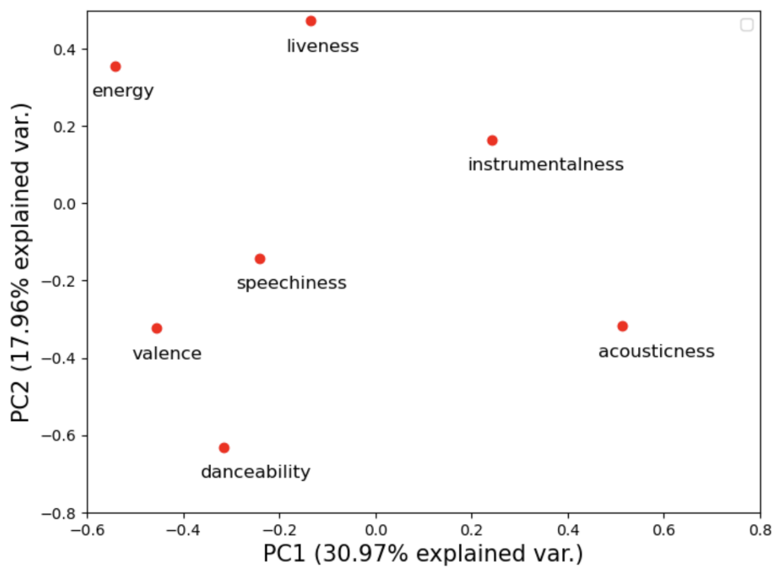



Fig. 2 Latent representation of audio features in first two PC

# 4 Methodology

## 4.1 Labeling

Labeling in our music recommendation system refers to the process of assigning context labels to each song in the dataset based on their acoustic characteristics. By labeling the music by context, we can recommend music to users based on their preferences.

Before we started labeling the music, we first came up with a list of contexts we wanted to categorize. Our labels are encoded as 0-focus, 1-relax, 2-energize, and 3-commute. These labels can be used to help classify and organize the songs, as well as to make recommendations based on the user's preferences.

There are several ways to label a song, such as manual labeling and automated labeling. Manual labeling is based on a human's subjective analysis of the song's characteristics. However, it can be very time-consuming and it also seems impossible to manually label all the songs in our dataset. That is, we decided to go with automated labeling by using a function to extract and assign the label of each song in the dataset based on its audio features. This can be faster and more scalable, but may be less accurate than manual labeling. We defined each of the contexts as follows, as shown in Table 2.

Table 2. Conditions of each context based on musical features

| Context | Conditions |
|---|---|
| Focus | acousticness > 0.1, speechiness < 0.05, liveness < 0.15 |
| Relax | valence > 0.5, speechiness < 0.08 |
| Workout | energy > 0.6, danceability > 0.6 |
| Commute | the rest |

We first calculated the mean, maximum, minimum, 25th, 50th, and 75th percentiles of each acoustic feature. Based on the characteristics of each context, we categorized the songs into different contexts based on the values of their acoustic features. For example, as for the specific context of focus, high acousticness and low speechiness can be appropriate characteristics. This is because music with minimal vocals is often preferred for focus music. That is, we defined songs with above 50th percentiles acousticness, below 50th percentiles speechiness, and below average liveness as focus.

**4.2 Cosine Similarity**

For our recommendation system, we used the cosine similarity algorithm to compute the similarity between pairs of songs. The cosine similarity is a measure of similarity between two non-zero vectors of an inner product space, and it ranges from 0 (no similarity) to 1 (perfect similarity).

To calculate the cosine similarity between two songs, we first extracted their audio features, such as loudness and danceability. We then represented each song as a vector of these features, where each feature corresponded to a dimension in the vector.

Our next step is to calculate the cosine similarity between each pair of songs by taking the dot product of their feature vectors and dividing it by the product of their Euclidean lengths, with equation (1) given below. This produced a similarity score between 0 and 1 for each pair of songs, where a score of 1 indicated perfect similarity and a score of 0 indicated no similarity.

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}} \tag{1}$$

## 5 Application Workflow

We developed a music recommendation dashboard using Streamlit that allows users to select a current context and customize their recommendations based on their desired acoustic features. The dashboard consists of several user interface elements that allow users to interact with the system and receive tailored recommendations.

The first component of the dashboard is the context selection panel. Users can select their current status from a list of options such as energize, relax, focus or commute, as shown in Fig. 3. The section allows users to communicate their current state of mind to the system.

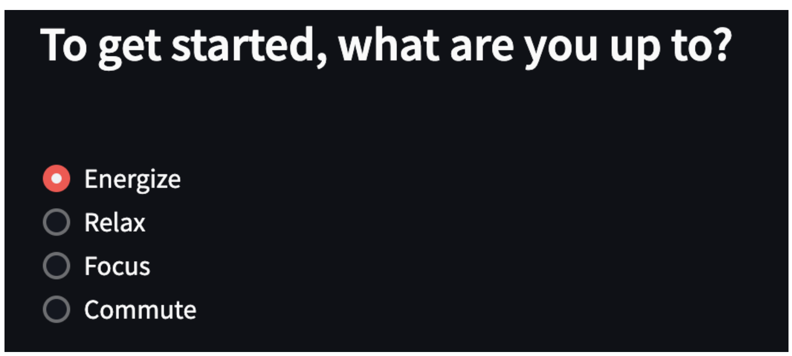



Fig. 3 Users can select their current listening context

The next section is the acoustic feature sliders. Those sliders allow users to customize their recommendations based on a variety of audio features such as danceability, energy, speechiness, etc. Users can move the slider to adjust the weight of each feature, with a range of values from 0 to 1 (Fig. 4). If users didn't make an adjustment, the default value will stay average for each feature. This feature enables users to refine their recommendations based on their specific preferences and context. In addition, if a user hovers the mouse over the question mark on the top right, it will show a detailed description of the audio feature.

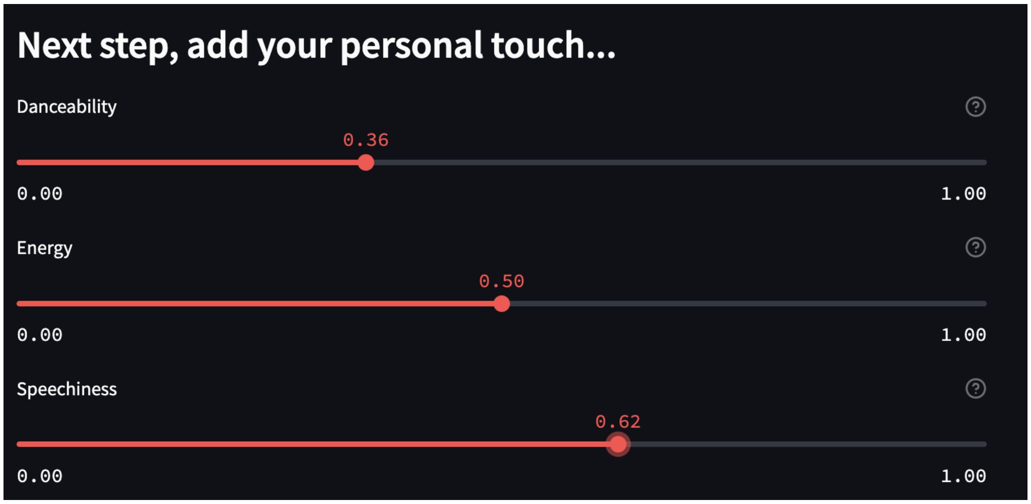



Fig. 4 Acoustic feature sliders that allow users to customize their preferences

The third component of the dashboard is the song selection panel. To this point, the recommender provides a list of 20 songs that match the user's criteria. Users can then select songs they like from the list, providing additional feedback to the system about their preferences.

Now, the recommendation system will generate a list of 10 songs that it predicts the user will like. The recommended songs panel displays the name and album cover for each song (Fig. 5). If the user clicks the name of the song, then it will be directed to the official Spotify url of that page where users can listen to the song.

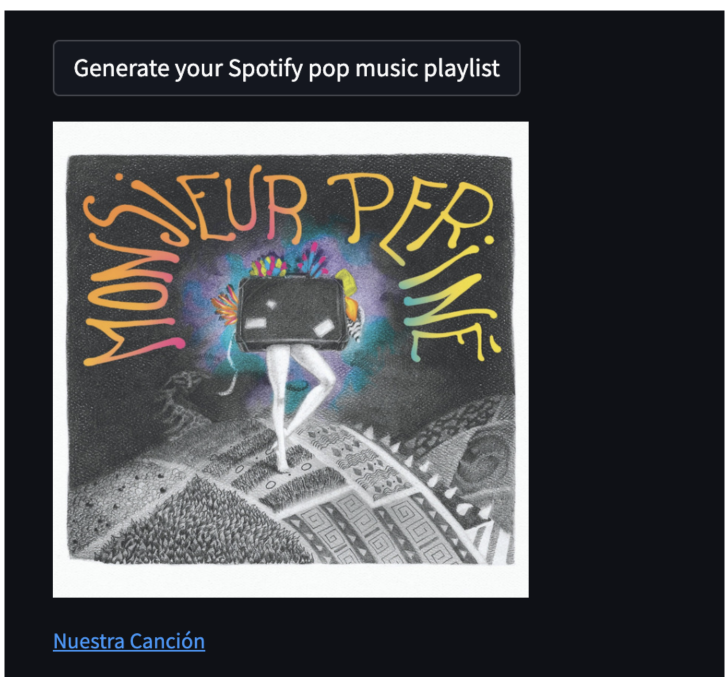
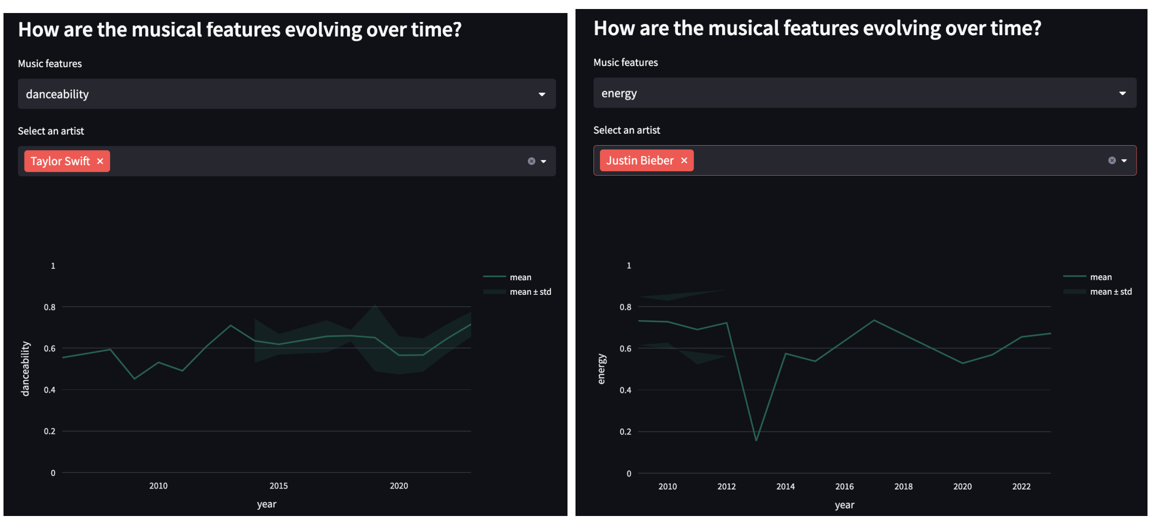



Fig. 5 An example of a recommended song from the playlist

## 6 Analysis and Discussion

From the pop music analytics dashboard, we conducted exploratory research on our data, where there are mainly three visualizations to provide insights.

First, the line chart we created of how music features have evolved over time can provide a clear visualization of trends and changes in musical trends and styles. It can also help to identify patterns and correlations between different music features and provide insight into the direction of future musical trends. For this visual analysis, we provide two filtering conditions, including the type of features and the top 10 artists with the most number of songs in our dataset. The center trace is the average value of the feature for the specific artist in that year, and the continuous error bands are calculated based on the mean ± one standard deviation. For example, Fig. 6 shows how the danceability of Taylor Swift's music changes over time, which has a slight shift from 2015 indicating her transition from country music to pop music. There are also some descending trends in energy during the pandemic for some artists, like Justin Bieber, when the tracks released at that time seem to be more peaceful.




(a) (b)

Fig. 6 How music features have evolved over time

(a) the danceability of Taylor Swift's music from 2006 to 2023

(b) the energy of Justin Bieber's music from 2009 to 2023

The second visualization is a graph showing the distribution of music features for certain artists' tracks. We can compare the differences in music styles by the selection of artists and music features. For example, in Fig. 7, Lana Del Ray's music has a unique alt-pop or baroque pop style, whereas her songs have relatively low speechiness. R&B and Dance Pop are Rihanna's main music genres, making her songs typically upbeat and catchy with medium speechiness. Hip-hop artists or rappers, such as Drake and Lil Wayne, would have more tracks with relatively high speechiness.

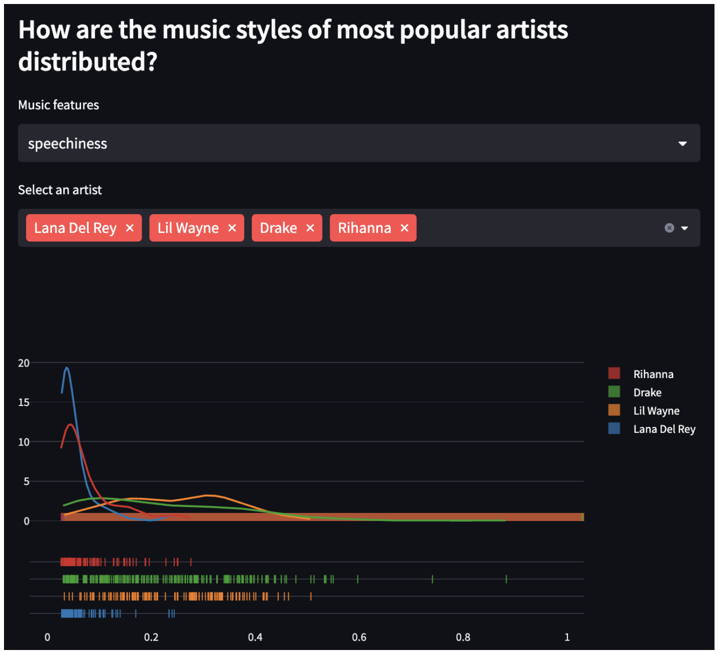



Fig. 7 The distribution of speechiness for 4 most popular artists' tracks

Finally, we created a visualization of how the popularity of a track is related to its musical features over all the tracks in the dataset as shown in Fig. 8. The correlation matrix between features and popularity is calculated and displayed using a heatmap. From this visualization, we are able to see that popularity does not seem to have a significant correlation with all music features. However, the energy level of a track has a relatively strong negative correlation with acoustics with a correlation coefficient of -0.67. Meanwhile, valence is positively correlated with danceability and energy, with a correlation coefficient of 0.43 and 0.40.

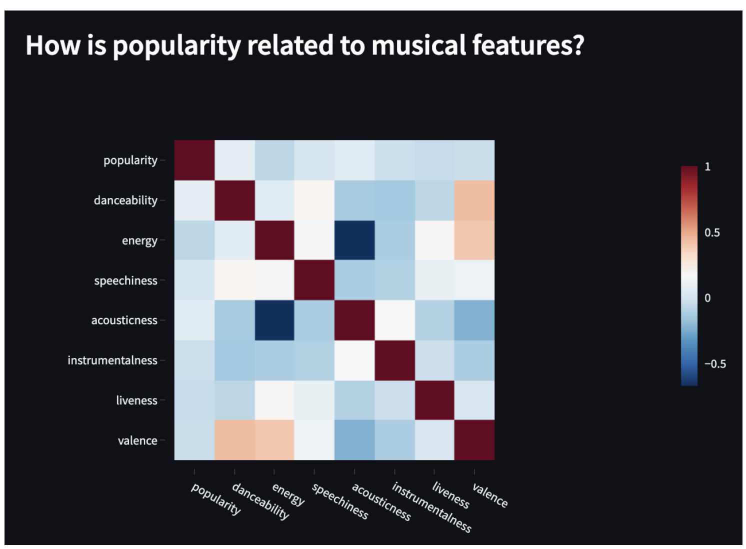



Fig. 8 The correlation between multiple music features and popularity

## 7 Conclusion

In this project, we preprocessed the spotify data by standardizing all the feature data, removing unnecessary features like loudness based on the MAD, filtering prioritized features such as energetic and acoustic features. Based on the findings we found in the preprocess procedure, we labeled the music and encoded into focus, relax, energize and commute. We utilized the cosine similarity to provide the music recommendation for the users.

In addition, we developed a music recommendation dashboard using Streamlit where users can find how features evolved over years given music features and specific artists' songs. Besides, users can choose different contexts and customize their preference based on the different features, and meanwhile the system will generate songs randomly and by choosing songs that match users' tastes, people can get a playlist of 10 new songs.

In summary, our final dashboard not only provides viewer's to find their liked singers' songs evolve over time and how the music will evolve based on different features, but also viewers can get songs based on their activities and contexts to enjoy songs they liked. In the future, we can evaluate our system by asking for users' feedback, including their ratings and attitudes for the tracks in the generated playlist. Besides, it might be beneficial to try out other recommendation algorithms, such as Spotify Web API's built-in "Get Recommendations" method to improve our recommendation results.

## Appendix

We have structured this project using a GitHub repository, and the system is deployed into a Streamlit application. Ruolin is responsible for data collection, data preprocessing, exploratory data analysis & visualizations, Streamlit dashboard development, user interface design, and Section 3 & 6 in the report. Mu conducted the literature review, model implementation, data labeling, user interface design, and Section 4 & 5 in the final report. Zhiang was responsible for the literature review, data labeling, and Section 1, 2 & 7 in the report.

We have applied various skills in data manipulation, analysis, and visualization as well as knowledge of programming and statistics from SI courses like SI 507, SI 618, SI 671, SI 649, SI 568, etc. When mapping and labeling tracks with user context categories, we first tried to use unsupervised machine learning with PCA and K-means clustering. Unfortunately, the unsupervised classification does not seem to show an effective result and the clusters cannot be sensibly explained or distinguished by musical features.

For EDA & visualizations, we tried to provide insights and represent the patterns within the dataset in multiple ways, including box plots showing how music features have evolved over time and 2D and 3D scatter plots showing the clustering results, which can be found in our workbook from GitHub repository.

For the other tools, we have also tried to use Plotly for visualization. However, we found out later that this module is more complicated to utilize and it is more difficult to integrate our system as a web application so we chose to use Streamlit instead.

## References

Girsang, A. S., Wibowo, A., & Edwin. (2019). Song Recommendation System Using Collaborative Filtering Methods. Proceedings of the 2019 the 3rd International Conference on Digital Technology in Education. https://doi.org/10.1145/3369199.3369233

Han, B., Rho, S., Jun, S., & Hwang, E. (2010). Music emotion classification and context-based music recommendation. Multimedia Tools and Applications, 47(3), 433–460. https://doi.org/10.1007/s11042-009-0332-6

Kathavate, S. (2021). Music Recommendation System using Content and Collaborative Filtering Methods. International Journal of Engineering Research and Technology, 10(2). https://www.ijert.org/research/music-recommendation-system-using-content-and-collaborative-filtering-methods-IJERTV10IS020071.pdf

Mavuduru, A. (2021). How to build an amazing music recommendation system. Medium. Retrieved April 12, 2023, from https://towardsdatascience.com/how-to-build-an-amazing-music-recommendation-system-4cce2719a572

Niyazov, A., Mikhailova, E. A., & Egorova, O. (2021). Content-based Music Recommendation System. 2021 29th Conference of Open Innovations Association (FRUCT). https://doi.org/10.23919/fruct52173.2021.9435533

Omowonuola, V., Wilkerson, B., & Kher, S. (2022). Hybrid Context-Content Based Music Recommendation System. Lecture Notes in Networks and Systems, 121–132. https://doi.org/10.1007/978-3-031-18461-1_8

Paul, D. J., & Kundu, S. (2020). A Survey of Music Recommendation Systems with a Proposed Music Recommendation System. Advances in Intelligent Systems and Computing, 279–285. https://doi.org/10.1007/978-981-13-7403-6_26

Pichl, M., & Zangerle, E. (2021). User models for multi-context-aware music recommendation. Multimedia Tools and Applications, 80, 22509-22531.

Srebrenik, B. (2021, December 6). Introduction to Music Recommendation and Machine Learning. Medium. https://medium.com/@briansrebrenik/introduction-to-music-recommendation-and-machine-learning-310c4841b01d#:~:text=A%20content-based%20approach%20is%20often%20the%20technique%20relied,about%20a%20particular%20song%20when%20making%20a%20recommendation.