SENG474 Formal Project Proposal: Hybrid Music Recommendation System

Nathan Pannell, David Kim, Yule Wang, Weiting Ye,
Adithyakrishna Arunkumar, Abhay Cheruthottathil

February 13, 2025

1 Introduction

Music streaming services have revolutionized the way audiences engage with music, with 90 million paid subscribers in the US alone. These platforms generate 89% of the industry's revenue, emphasizing the importance of continued user engagement. However, current recommendation systems often fail to provide diverse and relevant suggestions, leading to dissatisfaction. Addressing these limitations is essential for improving user experience and maintaining competitive advantages in this growing industry.

Our research focuses on developing a hybrid recommendation system that combines collaborative filtering and content-based filtering. The goal of combining these methods is to utilize co-occurrence information to detect patterns in commonly grouped songs, while also gathering extra data by analyzing intrinsic song features like tempo, volume, and key. We aim to better understand the relationship between songs, and what that means in terms of their musical features.

The outcome for this project will be a music recommendation system that suggests songs based on an input playlist containing one or more songs. This is to be achieved through our experimental model design outlined below and measured by comparing to real playlist data. By incorporating both user preferences and audio metrics, we aim to deliver a valuable algorithm that suggests diverse but relevant songs.

A key topic of this project is the lack of consistent and comprehensive data across our problem space. The primary contribution of this work, beyond the recommendation algorithm itself, is the potential to effectively use sparse data with major blind spots. This relevant as the largest music streaming platform in Canada, Spotify, is deprecating their API platform to cut down on public access to music data.

2 Problem Definition

2.1 Datasets

This project is utilizing two primary datasets to analyze patterns in track co-occurrences and audio features, separately. Both datasets contain Spotify track IDs for each distinct track, which will allow co-integration between the two. Additional datasets in this space may be considered, as the features are standardized from the (now deprecated) Spotify Web API. This allows for seamless concatenation of new track entries into the existing collection.

- **Spotify Million Playlist Dataset:** This dataset comprises 1,000,000 user-generated playlists, encompassing over 66 million tracks and more than 2 million unique tracks. For each entry, which represents a playlist, the name and follower count are available. Additionally, each track has title, artist, album, and duration attributes. This data provides insights into user listening behaviors and track co-occurrences. However, the feature data for each song is limited, so this will primarily benefit the recommendation model by forming relationships between songs based on playlist membership.

- **Spotify 1.2M Songs Dataset:** Containing audio features for approximately 1.2 million tracks, this dataset includes 24 attributes including audio features and metadata. There are 11 metadata features including title, release date, and artists. The 13 audio features are consistent with the Spotify Web API and include values for danceability, key, acousticness, and more. Notably, it primarily consists of randomly sample, less popular songs, resulting in limited overlap with the Million Playlist Dataset. Nonetheless, these audio features are objectively tied to the songs, so they can be used for non-user-based suggestions informed by track content rather than listening behaviour.

2.2 Risks and Backup Data Sources

The primary risk to success lies within the limited overlap between the two datasets. As mentioned, the songs for which we have audio features are sampled at random from Spotify's 100 million songs, and thus may poorly represent the 2 million distinct songs from the Million Playlist Dataset.

As such, the first step of the data analysis phase will consist of determining this overlap and trimming the datasets to contain playlists with strong overlap (i.e. the playlists for which we have audio feature data for multiple tracks in the playlist.) To mitigate this issue, we plan to implement the following measures to ensure robust data:

- **Independent Training:** Since each dataset is large and complete on its own, we can train individual models on each separately for partial recommendation systems. For example, we can create an autoencoder for the feature data without worrying about overlap for playlist data. We can also build a track ID-only recommendation model based on the Million Playlist Dataset without needing audio features.
- **Trimming Sparse Datapoints:** When combining the two datasets, we can take advantage of the large scale of both to decrease the size, but increase the quality. For example, if we take the top 1% of overlapping playlists with the most data, that still represents a 10,000 sample dataset which may be all that's required to build an accurate algorithm.
- **Proxy Features:** If the overlap issue presents too much of a challenge, we can investigate augmentation methods to estimate song features based on relationships to other tracks for which these attributes are known. It should be noted that this is a last resort option as it will decrease the accuracy of our input data, but it may be required if the overlap is insufficient.
- **Alternative Datasets:** Actively exploring additional datasets like the Million Song Dataset or the Free Music Archive (FMA) can add more samples to our data model or potentially add a third axis for analysis. Additionally, there are other datasets available on Kaggle.com that contain the same feature information from the Spotify Web API such as the Top Spotify Songs of 2023 or the Spotify Tracks Dataset that could be integrated with our audio feature collection and potentially contain valuable, popular tracks.

3 Goals

Our primary objective is to develop a hybrid recommendation system that effectively combines collaborative filtering and content-based filtering to provide accurate song recommendations. Success will be measured through:

- **Ranking Performance:** We will evaluate how effectively the model generates an ordered list of recommendations by measuring the position of a known track (or known tracks) from the original playlist within the ranked predictions. Our goal is to minimize the average position where relevant tracks appear in the ranked list. For example, a playlist containing 50 songs could feed the first 40 into our recommendation system, then see how far down the list of suggestions one of the 10 holdout songs is positioned.
- **Partial Playlist or Single Input:** The recommendation system should be able to generate suggestions in the same manner for an individual song or a playlist of multiple songs. To achieve this, we must expose the model to inputs of varied sizes so it doesn't bias towards small or large inputs. This can ensure the system is useful in more applications than just sequence completion or individual track co-occurrence identification. Partial playlists are evaluated as mentioned previously, however individual track suggestion accuracy can be evaluated by identifying common songs across unseen playlists and checking if the new co-occurrences are predicted by the recommendation system.

4 Plan

4.1 Experimental Approach

The specific experimental approach for this project will be left open-ended as more work is still required to identify the data overlap and potential pain-points regarding input data. Once this has been solved, there are a few techniques that could be helpful to build the model:

- **Dimensionality Reduction:** The 24 features (13 of which are audio-based) represent a large space of song representations. Techniques like Autoencoders (AEs) and Principle Component Analysis (PCA) could compress the representation of these tracks down to a small dimension which make interpretation and navigating feature space more feasible.
- **Feature Engineering:** Given a featureless track (not included in the audio features dataset but included in the playlist dataset), it is possible to approximate its features using a weighted combination of attributes in related songs. This could include approximations based on genre, artist, or album; it could also involve identifying playlist-based relationships and determining a consistent thread between songs that are frequently listened to together.

4.2 Task Breakdown

Each team member will be involved in a machine learning aspect of the project:

- **Data Integration, Exploratory Analysis and Feature Engineering:** (Yule Wang and Weiting Ye) Yule and Weiting will merge the Spotify Million Playlist Dataset with the Spotify 1.2M Songs Dataset using track identifiers and perform initial data analysis on the combination dataset. After the pre-processing is completed, this team can develop proxy features for tracks lacking explicit audio features using metadata. It will be a goal of this team to quickly generate a usable data model for the other teams' use, then to build a more comprehensive understanding (potentially including feature engineering) for second or third passes of the model training stages.
- **Initial Model Creation:** (Adithyakrishna Arunkumar and Abhay Cheruthottathil) After the data has been cleaned and prepared, Adithyakrishna and Abhay can design and train the initial models. This will consist of models based on each dataset separately, as well as the combination dataset provided by Yule and Weiting. This team will be responsible creating the underlying models that have high prediction accuracy based on the distribution of the data. These models will be thoroughly tested, then integrated into a final system by the next team.
- **Model Integration, Evaluation and Analysis:** (David Kim and Nathan Pannell) David and Nathan are responsible for combining the collaborative and content-based models into a unified hybrid recommendation system. This will involve using the models' predictions to create effective and accurate recommendations, beyond merely fitting the existing data. This will also be where the models are evaluated on a series of metrics, and tuned for optimal recommendation generation.

4.3 Timeline

- **Initial Data Integration and Pre-Processing:** Complete by February 24, 2025.
- **Preliminary Hybrid Model Creation:** Complete by March 10, 2025.
- **Final Recommendation System Completion:** Complete by March 24, 2025.

NOTE: This timeline is tentative and will be adjusted as the project evolves.