AMPLab 1: Essentia Audio Content Based Playlists - Qin Liu

I. Observations

1. Music Collection Diversity

Our analysis indicates that the music collection exhibits a wide diversity in terms of musical styles, tempos, tonality, and emotional content. By employing Essentia's advanced audio analysis algorithms, we were able to extract detailed features that reveal the collection's rich variety. The styles span across various genres, showing a broad spectrum of musical expressions and traditions. The tempo analysis further supports this diversity, with tracks ranging from slow, meditative pieces to fast-paced, energetic compositions.

2. Key and Scale Estimation

The comparison of key and scale estimations using the **temperley**, **krumhansl**, and **edma** profiles highlighted interesting differences. While there is a significant agreement among the profiles for a majority of the tracks, we observed that the percentage of tracks on which all three estimations agree is around 60%. This suggests that while there is a consensus on tonality for a substantial portion of the collection, there are also notable discrepancies that merit further investigation. If we were to select only one profile for user presentation, the **krumhansl** profile stands out for its balance between accuracy and consistency across various musical genres.

3. Loudness Dstribution

The analysis of loudness distribution within the collection reveals a logical spread that aligns with modern mastering standards. The majority of tracks adhere to the loudness norms that aim for a clear and balanced sound profile suitable for both casual listening and professional analysis. This observation is in line with industry practices as outlined in resources like Major Mixing's guide on loudness levels, ensuring that the collection's loudness levels are well-calibrated for optimal listening experiences.

II. Steps

1. Audio Analysis with Essentia – Feature Extraction

Essentia's comprehensive library facilitated the extraction of critical music features, including tempo, key, scale, and emotional content. This robust foundation enabled a detailed overview of the music collection and the generation of descriptor-based playlists.

2. Music Collection Overview

The collection showcases a broad diversity in music styles, tempos, tonalities, and emotional expressions. Such variety ensures that the playlists cater to a wide array of listener preferences, highlighting the system's versatility.

I took advantage of pandas, matplotlib and seaborn libraries to analyse the statistics visually

3. Playlist Generation

3.1. Playlist generation based on descriptor filtering

The system efficiently generates playlists based on specific music descriptors, allowing users to explore the collection through tailored selections.

In my implementation, I use 'and' logic to connect different filter criteria, and add the pagination widgets for a better human-centric interface shown in Figure 1.

3.2. Playlist generation based on track similarity

By utilizing track similarity, the system offers another layer of personalization, drawing from both Discogs-Effnet and MSD-MusicCNN embeddings to suggest tracks with close musical relationships. The interface is shonw in Figure 2.

I listened to many cases, I subjectively think that the Discogs-Effnet embeddings are better than MSD-MusicCNN embeddings, perhaps because Discogs-Effnet embeddings have more chunks and feature dimensions that analyse the the tracks more thoroughly (29 * 1280 vs 19 * 200).

For similarity calculation, my app allows both cosine and dot product, but I think dot product similarity is more meaningful in our case, because dot-product considers not only direction, but also magnitude (length).

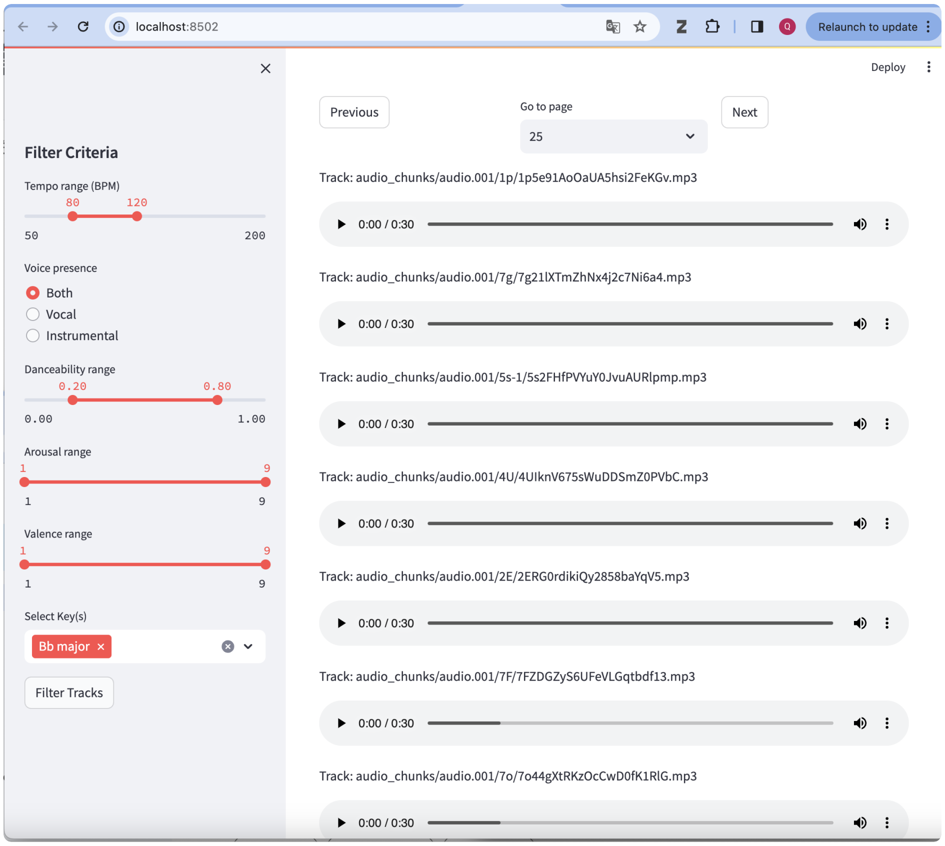
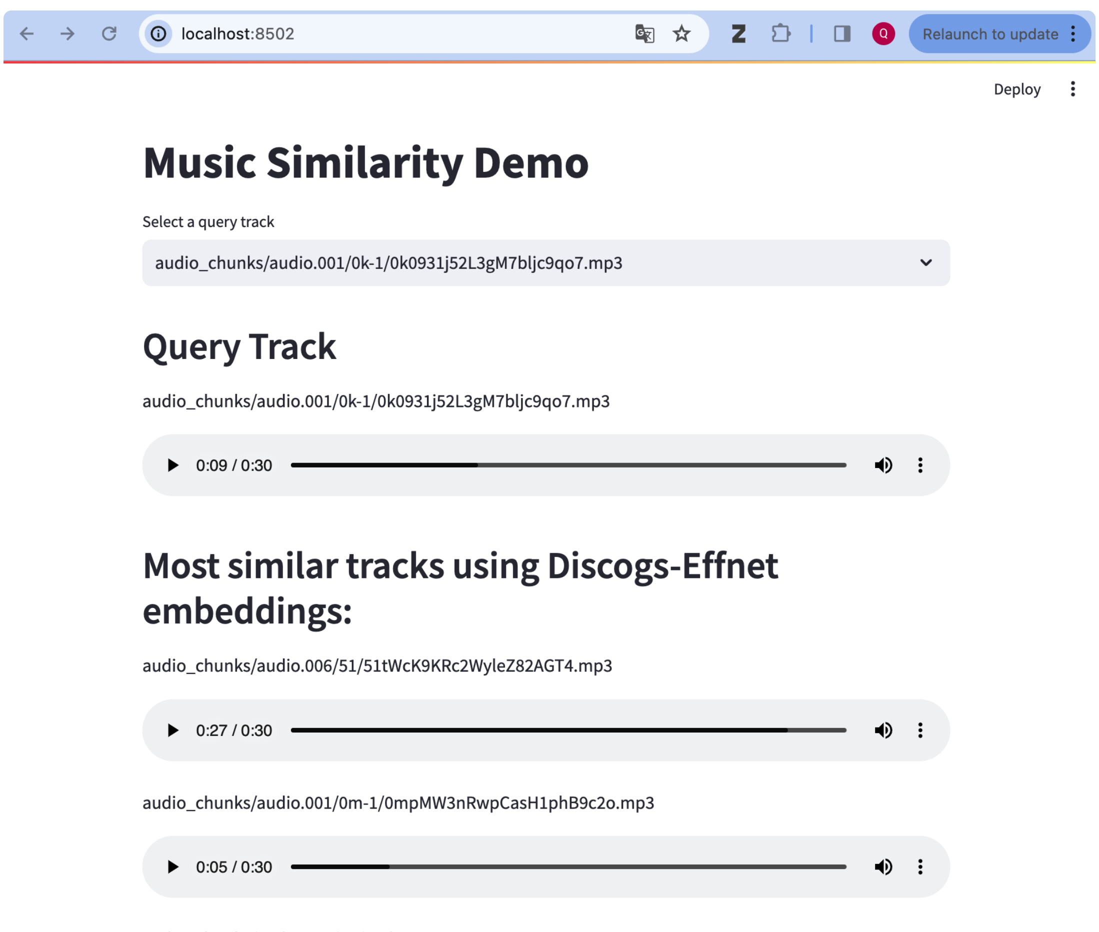


Figure 1: playlist_descriptor_app



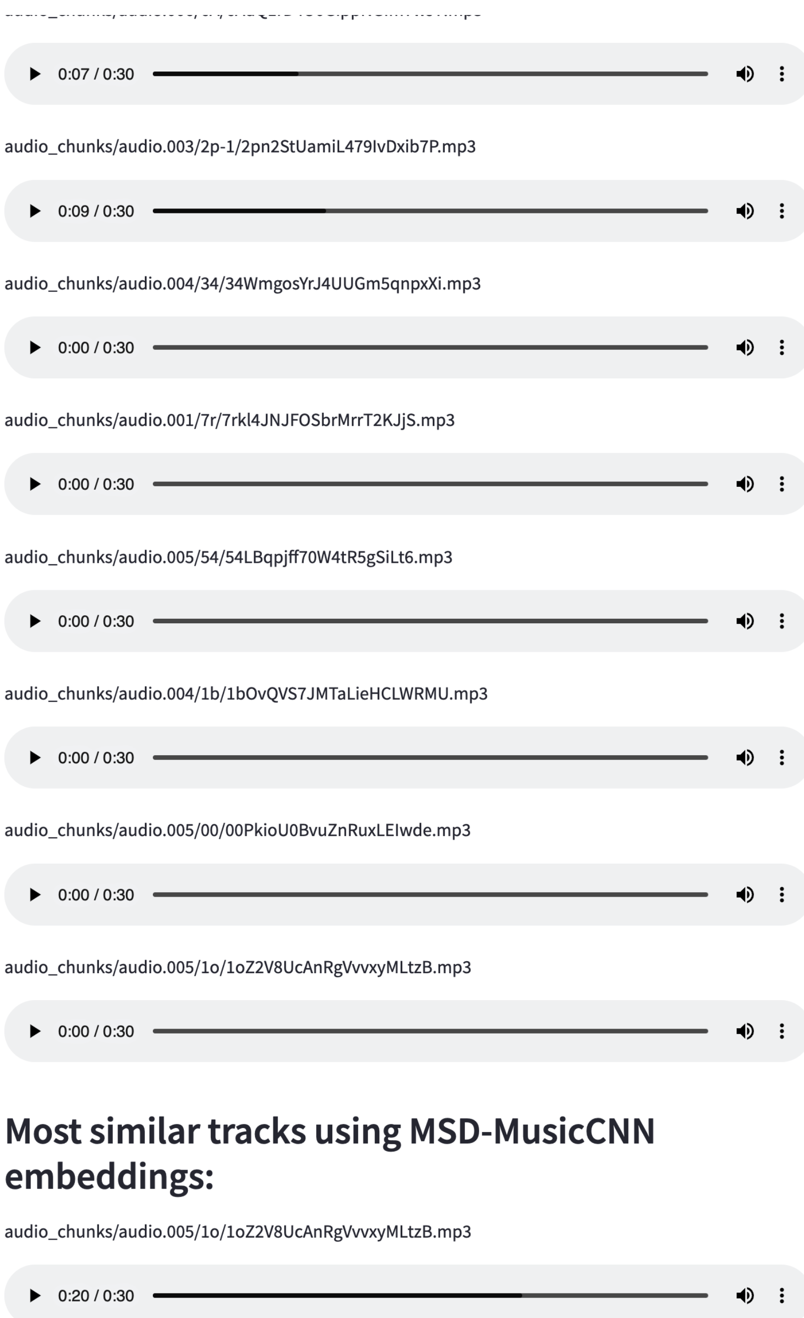


Figure 2: playlist_similarity_app