Department of Computer Science, Memorial University of Newfoundland
COMP3100: Web Programming
Dr. Amilcar Soares
February 4th, 2022

Project Name: Iteration One

Team 10:
Nguyen, Nhu N (201916426)
Chicas Dueñas, David (201919354)

Introduction

As music consumption shifts toward online platforms, automated playlist continuation is becoming a more and more important problem in music recommendations. For many users, compiling playlists manually is highly time and effort consuming since this can be a difficult task given the diversity of tastes and the large amount of musical content available.

This project proposal will look into the task of adding one or more tracks to a playlist so that the intended characteristics of the original one remain the same. Different methods have been proposed depending on several factors such as playlist titles or the number of songs in the playlist. To achieve this goal, we plan on using several datasets such as the Million Playlist Dataset (MPD): “Sampled from the over 4 billion public playlists on Spotify, this dataset of 1 million playlists consist of over 2 million unique tracks by nearly 300,000 artists, and represents the largest public dataset of music playlists in the world”. The purpose of this project is aiming to help users create and extend their own music playlists more easily, and ultimately help people find more of the music they love.

Proposal

We want to develop a playlist generator where clients can use the service to get a playlist with music specific to their likes. The user will answer some questions about their preferences in music like preferred genre, era, style of music, and then be presented with some sample songs to listen to. The user will then like or dislike the songs presented to them; this will help us narrow down the songs that will be used in the playlist. After the playlist has been created we want to let the user be able to add their playlist to their Spotify or Apple Music account. We also want to present some visualization of the statistics of the music presented to the client. For example, a graph showing what is their preferred genre to their least favorite, or how many times is the same artist repeated in the playlist and many other stats. We are planning on using the following third-party API’s.

- The Spotify API, which will help us present the user with music they might like and fetch that information from Spotify so that we do not need to download all the data.
- We will use the Apple Music API to do something similar to the other API and allow information from multiple sources to be presented to the user.

We are planning on using multiple data sets to have a broad range of songs available. One of them will be “Spotify Song Data Set” on GitHub by user rfordatascience and “Spotify Million Playlist Dataset” by Spotify. We hope that our project helps users find new music tailored to their taste and provide an accurate playlist of what they like. We think that the two datasets will give the user a wide range of songs available for both the sample songs and the playlist generator.

Functionalities

Main:

- Get info: Present a list of questions for users to answer and record the given inputs.
- Suggest songs: Present a list of songs using the previously recorded answers and ask the user to select the ones they like.
- Analyze playlist: Allow users to upload personal playlist and use that to narrow down relevant songs for recommendation.
- Create playlist: After reducing the song search space to a smaller set of candidates, generate a playlist using the user’s preference.
- Save playlist: Allow the user to name and save the created playlist to their own account.
- Add playlist: Give the user the option to add the playlist to their personal library in music apps such as Spotify and Apple Music.
- Visualization: Display statistics of the generated playlist using charts and graphs.
- Save visualization: Save the charts and graphs generated for the user.

Authentication:

- Registration: Get account username, email, password, and password confirmation from the user and create an account based on input after checking if they are valid.
- Login: Get account username (or email) and password from user and login.
- Forgot password: Get account username (or email). Send a link to the email for resetting the account’s password and update the password in the backend database.

References

ACM Spotify RecSys Challenge 2018 1st place solution. Zhaoyue Cheng | ACM Spotify RecSys Challenge 2018 1st Place Solution. (n.d.). Retrieved February 3, 2022, from https://joeycheng.me/projects/recsys_challenge_2018/

Spotify Million Playlist Dataset Challenge: Challenges. AIcrowd. (n.d.). Retrieved February 3, 2022, from https://www.aicrowd.com/challenges/spotify-million-playlist-dataset-challenge#dataset