Spotify User Music Analysis

Author: Aswin Yayathi Andal

Project Description:

This project aims to predict user music preferences and check if a new song will cater to the user’s taste based on the songs in his library(excluding his listening history from other platforms).

Simply put, we’re checking if a user will listen to a song, by understanding his taste in music by leveraging two important concepts:

1. Spotify’s API to retrieve track data
2. KNN Algorithm(to predict user data)

This document is a detailed summary of the project aimed to provide a detailed documentation of the processes involved, the topics it touches on to hopefully give the reader a fair understanding of the project.

Sptofy’s API:

Spotify provides developers an option to download their data and retrieve more details about the artists, albums, tracks using the ‘id’, by allowing users to send requests through it’s API to pull data from it’s servers in a JSON format.

In this project, we require the details of a track to understand the user’s listening patterns, the kind of music he/she listens to and train the model based on those tracks, as well as other tracks.

The features that we retrieve using Spotify’s API are:

1. danceability
2. energy
3. key
4. loudness
5. mode
6. speechiness
7. acousticness
8. instrumentalness
9. liveness
10. valence
11. tempo
12. type
13. id
14. uri
15. track\_href
16. analysis\_url
17. duration(in milliseconds)
18. Time signature

To retrieve the user’s data first, we need to request spotify for the user’s data, which Spotify allows the user to download in JSON format, which has to be cleaned and transformed using Python(Pandas), converted a dataframe and then the track id is sent to the API to retrieve the track features. This process has to be repeated for every track in the user’s library.

The next step after retrieving the data, is to load all the track features into a dataframe, remove unnecessary attributes that aren’t required to train the model.

Post that, the model is fitted and transformed and scaled(since this algorithm measures distance b/w data points) and the model is trained accordingly.

The model is split into a 60:40(train:test) model and trained, after which the f1 scores are calculated.

Upon training the model, the training set scored 85%, while the test set scored 81%. While the difference is not high, the model still has room for improvement.

SCOPE FOR IMPROVEMENT:

1. It was observed that spotify doesn’t identify the language for each track and having that data can provide room for investigation.
2. Other algorithms can be explored and be compared to the KNN algorithm.