Listen to Data

Spotify Data Analysis and Exploration

INFX 547 A - Social Media Data Mining and Analysis

Yu Wang | Wuyi Zhang | Tapasvi Bansal

Final Project

June 2017

Note

We put all our work in a GitHub repository (Link) where information and results of the data we collected, cleaned, and analyzed can be found. Also, most of our discussions and interpretations of the results are presented along with the visualization in Tableau (Link).

Background

Music, essentially, is who we are (Philpott & Plummeridge, 2001). Our group is curious about what people listen to with Spotify. It is possible that if this project reviews Spotify data over time, we will be able to see changes of people's taste in music.

Research Question

The most popular artists on Spotify in 2016 are Drake, Justin Bieber, Rihanna, Kanye West and Twenty One Pilots (Spotify, 2017). We want to know what they have in common and what features in their songs may have contributed to their success. Also, we want to investigate corresponding aspects of their related artists. Moving into 2017, we wonder if the new releases continue the trend from 2016 or if there are emerging new genres. Specifically, we raised four questions to answer when analyzing our data:

- Most common genres for 2016
    - 5 most popular artists
    - 25 of their related artists
- Duration, explicitly, and popularity for 2016 top artists' top tracks and their related artists' top tracks
- In track features
    - Average danceability
    - Energy
    - Loudness
    - Liveness
    - Acousticness
- Most common genres for new releases

Data Collection

In general, we tried to answer those questions via analyzing data from Spotify. To collect and analyze data, we use Python as programming language for scripting.

We utilized official Spotify REST API. Also, we used a lightweight Python library Spotipy, which comprised Spotify Web API, for easy authentication.

There were two situations when sending requests from client to those RESTful APIs. For non-authorized request, we merely needed to create a Spotify object and make method calls with the parameters as we like. For example, we could specify the artist name and their music styles as terms for searching. For authorized requests, which required user authentication when calling methods, we needed to generate authorization tokens and register our applications to get the credentials necessary to make authorized calls.

With different endpoints, we majorly focused on collecting the data we needed to answer the questions raised above. Specifically, we used its client module to collect data. By creating Spotify API objects and specifying relevant parameters, we collected data about albums, artists, and their top tracks as well as similar artists, their popularity, and their album categories. Also, we collected tracks' audio features, which indicated the detailed style information of the tracks.

We used the native Spotify REST API to extract artist data, track data, track feature data, new releases, and other related artists' data. We also used a Python package, Spotipy, when working with authentication. It provided us a way to seamlessly manage credentials in a project that heavily relies on collaboration.

Data Cleaning

The official Spotify API are RESTful services; the responses are in the format of JSON files. We did not need to build any pipeline for data persistence. We stored the JSON files locally for data processing.

After obtaining all the data and store them in JSON files, we conducted data cleaning. The primary goal of this process was to extract the information we needed among metadata stored in JSON files, which was represented in dictionaries in Python. In this step, we extracted certain parts of the data iteratively using specific keys and loops. We saved cleaned data in Pandas dataframe when conducting grouping and aggregation analysis.

Analysis Method

Overall, we have data for the 5 most popular artists of 2016; each of them has 5 related artists (25 related artists in total); 10 top tracks of each artist above (250 tracks in total); all the features of each track (250 feature files); and 20 new album releases of 2017.

We started with Spotify's 5 most popular artists of 2016 (Spotify, 2017). We extracted data of their top tracks and the respective features. Then we branched out our search to include the related artists of the most popular artists to create a network of artists to induce a wider range of diversity in our analysis. A diverse dataset is useful to help produce more accurate results. On the other hand, all these artists share much in common to be connected to each other. This prevents too much anomaly to a degree.

For the data analysis process, we used the Pandas, JSON, and Matplotlib packages to unpack the JSON files and start to explore the variables and their relations with each other. This allowed us to create visualizations with targeted variables based on the relations.

Ethical Consideration

We only extracted data obtainable with or without a basic Spotify developer account. No user data of any sort was collected. We consider this data collection and analysis to be very low risk with few ethical implications.

Limitation

Spotify API offers a vast amount of data from artists, tracks, users, playlists, each with very detailed granularity. We only explored the artists and their immediate related artists and tracks. We didn't take location or specific user habits into consideration, which is a missed opportunity to analyze how and why certain artists and tracks appeal to users. The statistical concepts in our analysis are more descriptive than predictive. This leaves room for more statistical exploration.

Visualization

The visualization tool of choice is Tableau for its versatile ability to create graphs and tell a story. The visualization and the web version of this report is hosted in the (Link).

Links

Web Version: https://thegreatgiraffe.github.io/Spotify-Data-Analysis-and-Exploration/

GitHub Repo: https://github.com/TheGreatGiraffe/Spotify-Data-Analysis-and-Exploration

Citation

Spotify. (2016, December 07). Drake is Spotify's Most Streamed Artist for 2016. Retrieved June 04, 2017, from https://news.spotify.com/us/2016/12/01/wrapped-2016/

Philpott, C., & Plummeridge, C. (2001). Issues in music teaching. Pg. 48 - 51. London: Routledge/Falmer.