Final Project Proposal - SI507

Author: Haolin Li

Email: haolinli@umich.edu

Last Updated: 10/30/2023

Github Repository: https://github.com/HumblePasty/SI507

Introduction

The idea of the final project originates from a former project I worked on:

- MusicMap - Github Repository
- Webpage

This project is a web application for mapping all kinds of music data. We created multiple visualizations like story maps for traditional music along the Silk Road, music festivals in China, and static pages of introductions etc.

In the "Music Globe" section, I implemented an interactive webpage based on **ArcGIS Maps SDK for JavaScript** to show the trends and most popular artists around the world. The music data comes from MusicBrainz Database.

Overall Goal: This project mainly focuses on presenting the data and lack of analysing. For the final project, I want to continue to work on this project and implement the analysis like network analysis on artists.

Data Source Description

MusicBrainz is an open music encyclopedia that collects music metadata and makes it available to the public. It is similar to the freedb project, but with richer data and a more advanced underlying structure that allows for better data retrieval and understanding. The primary features of MusicBrainz include:

- **Rich Metadata**: Information about artists, albums, tracks, and labels.
- **Relational Data**: Data on collaborations, band members, aliases, and more.
- **Acoustic Fingerprints**: Links to AcoustID, which lets users identify music by the actual sound.
- **Web API**: Allows developers to query and retrieve data programmatically.
- **User-driven**: Users can contribute, edit, and vote on metadata submissions.

Project Goal

Data Fetching

- Use the **MusicBrainz Web API** to fetch data. Given that some requests might be rate-limited, consider caching previous request results to ensure you're not repeatedly fetching the same data.

- Gather information about artists, **particularly about their collaborative projects**.

Note: In the Project Overview pdf **cawling webpages** is mentioned. MusicBrainz provides a well structured and handy WebAPI so right now I cannot think of whether it is necessary to use crawling technique.

Data Structure and Analysis

- Utilize a **graph** to represent relationships between musicians.
  (Each artist is a node, and collaborations form the edges.)

- Apply network analysis techniques like **centrality, community detection,** etc., to identify the most influential musicians or the largest collaboration clusters.

Data Presentation, User Interface and Interactions

- Use **Flask** as the backend framework and **Vue** as the frontend framework to create a **web application**.

- Integrate graphical libraries, like D3.js or Cytoscape.js, to **display the network of musicians** and allow users to visually perceive connections between artists (possibly interact with the networks and navigate through artists etc.)

- Allow users to **search for specific artists** and highlight their position in the network.

- Permit users to opt for viewing specific types of relationships, such as collaborations from a particular decade or within a specific music genre.

- Provide data filtering and sorting functionalities, such as by the number of collaborations, popularity, etc.

Other Function or Extension (time permits)

- Incorporate the former Music Globe webpage into the new web application and allow navigation through pages by adding tabs for example.

- Add a detailed page for each artist, showcasing their basic info, albums, tracks, and collaborators.

- Implement a recommendation system approach, suggesting other artists with a similar style or frequent collaborators when a user views a particular musician.

Appendix: Data Example

Artist Data for Pink Floyd