Final Project
The Mix Tape Map

Team Members:

Val Denay Mack (dgm97)

Sung Kyun (Kevin) Park (sp948)

Yuxiang Peng (yp344)

and

Wodan Zhou (wz262)

0. Introduction

Our project is the Mix Tape Map. This recommender system with user interactive design inspired by the capstone design platform, is based on the music recommendation system. Although this class is data-driven visualization class, we might want to understand the essential part of the recommendation system. From the Wikipedia, The recommender systems typically produce a list of recommendations in several ways. 1) through collaborative 2) content-based filtering and 3) the personality-based approach. The differences between collaborative and content-based filtering can be demonstrated by comparing two popular music recommender systems, Since there are several well-known music recommendation system outthere, Last.fm and Pandora streaming music service are popular music recommendation systems based on their data-driven recommendation algorithm. Last.fm creates a "station" of recommended songs by observing what bands and individual tracks the user has listened to on a regular basis and comparing those against the listening behavior of other users. Last.fm will play tracks that do not appear in the user's library, but are often played by other users with similar interests. As this approach leverages the behavior of users, it is an example of a collaborative filtering technique. Pandora uses the properties of a song or artist (a subset of the 400 attributes provided by the Music Genome Project) in order to seed a "station" that plays music with similar properties. User feedback is used to refine the station's results, deemphasizing certain attributes when a user "dislikes" a particular song and emphasizing other attributes when a user "likes" a song. This is an example of a content-based approach. Each type of system has its own strengths and weaknesses. In the above example, Last.fm requires a large amount of information on a user in

order to make accurate recommendations. This is an example of the cold start problem, and is common in collaborative filtering systems. While Pandora needs very little information to get started, it is far more limited in scope (for example, it can only make recommendations that are similar to the original seed). Just like these Last.fm and Pandora, we can also see these types of music recommendation services from Apple iTunes, Amazon Prime Music, YouTube Red, Spotify and TIDAL. Since one of the key functionality of the recommendation services is helping users discover items they might not have found by themselves, the recommender systems are a useful alternative to search algorithms. Interestingly enough, recommender systems are often implemented using search engines indexing non-traditional data. Getting initial inspiration from these music recommendation services, our project, the Mix Tape Map is aiming to have initial functionality of recommendation system. By doing so, we hope we could get a good inspiration of designing, producing, and evaluating final deliverables to develop our creativity, practical skills, teamwork and leadership.

1. Motivations

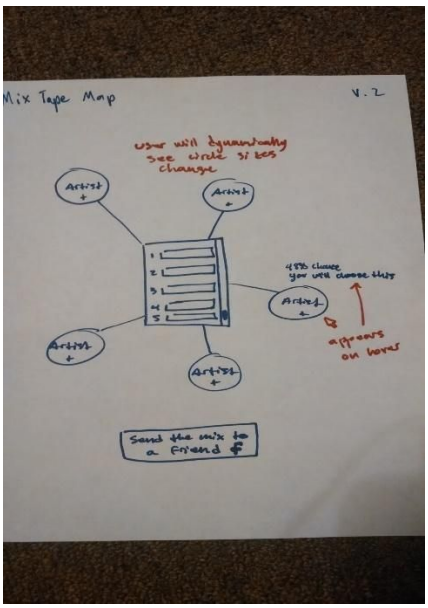
Listening to or discovering new music was a challenge on users' side. With massive developing of internet speed all over the world, the internet music listeners now can have in their hands such a vast music catalogue that we probably never imagine before. There are many of online music streaming services that offer their own music recommendation system. Some services focus on generation of playlists, others might focus on expanding their music catalogue even further, or focus on a more personalized music recommendation list. These systems present their results to their internet music listeners usually with a list style or a grid design of music artists. However,

lists or grids might not be the best way to give the music explorers enough information about the correlation in-between the recommended results. From this initial question, we would like to propose a possible solution to represent the artists' similarities as a network of inter-connected artists with using of line graphs, where a node is a music artist, and each edge in-between them represents a strong connection as well as an interactive design with different radius.

2. Work Completed by Each Team Member

2.1 Val Denay Mack (dgm97)

From the Skype meeting with team members, Val proposed the music recommendation system with an interactive and dynamic design. We might see these types of recommendation system from Spotify, iTunes, Soundcloud and other music streaming sites.



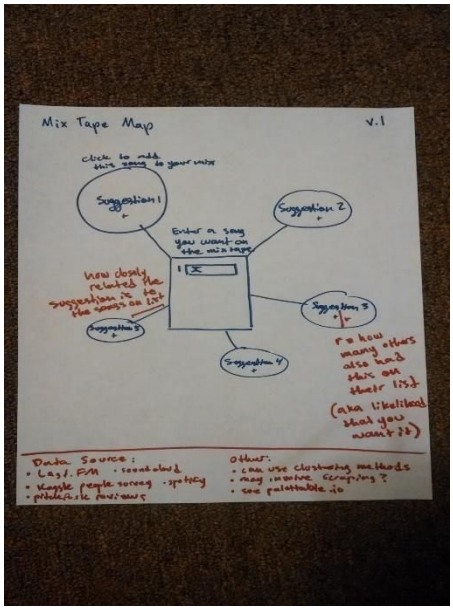


Figure 1 and 2: Early sketches of the Mix Tape Map

The basic concept of this visualization is that it uses data on music similarity and popularity in

order to dynamically show what groups of songs users are likely to be most interested in. In this context, the user can take advantage of that information by creating a mix tape for themselves. The user inputs the name and artist of a song that they like, and we populate the top 5 songs that are ‘recommended’ for them. Recommended just means that the largest number of people also liked the song we are suggesting, and based on the Spotify API, it is related to the initial song selection. Val also looked through Spotify API source to see if it is a good solution for us. Set up Node/Express web app and Spotify developer app to test API functionality and oauth. Ultimately we found that a web app was not necessary and we could collect all the required data through the use of AJAX. Also, Val provided the group with updates on how we can use the API to get information about co-occurrences of songs in playlists that will represent probability a given user will like a song. Val made a working prototype of the Mix Tape Map which has basic search functionality so that the user can search for an artist and a song will randomly be selected from that artist's top songs as the starting node.

Figure 3: Working prototype of the Mix Tape Map

The reference site is a design from Dribbble which has their own recommendation system with clean and intuitive design theme.

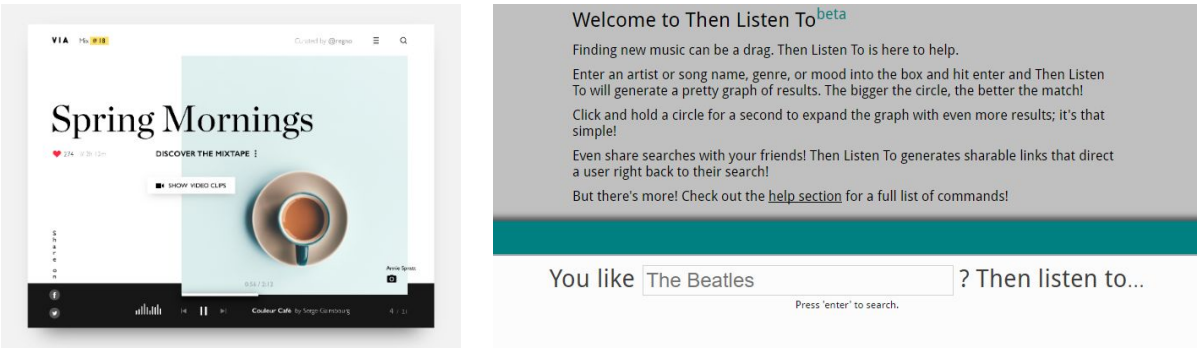


Figure 4 and 5: Inspired design theme from Dribbble and Then Listen To

The actual prototype has been built from the "Then Listen To" as the starting point. Val merged "Then Listen To" and "Music Recommendation" for the front-end design, using a D3 force layout. Our further plan was finishing design parts with interactive design components such as, changing radius of each node with actively changing distance lines. We could not finish the work this time, but we would like to have a good inspiration from this project that we can utilize it for the future projects.

2.2 Sung Kyun (Kevin) Park (sp948)

In terms of capstone design, designing an effective recommender system should be carefully considered with intuitive yet easy to understand design as well as the reliable recommendation process. In terms of its designing process, we were trying to get implemented from the concept of Capstone Design as we tried to express its functionality with a creative design with understanding the process from product development to production by carrying out the design

tasks of the product. The elements of Capstone Design can be proposed as establishment of objectives and criteria, synthesis, analysis, construction, testing and evaluation. Below is the flowchart of the Mix Tape Map prototype.

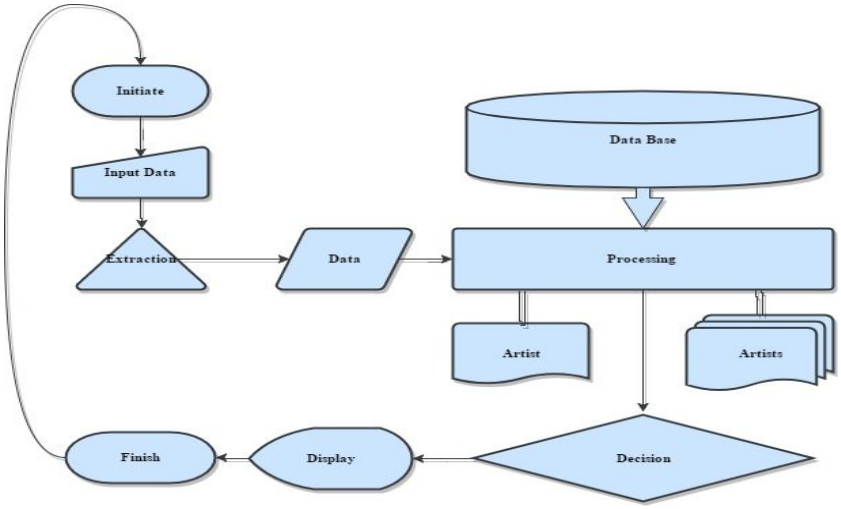


Figure 6: Flow chart of the porotype Mix Tape Map

The main assigned part for Kevin is data processing. Since we need to make sure feasible data set we can use on the prototype, Kevin mainly worked with Last.fm and Spotify API. Since the Last.fm has a strong data set we can use, the result from Last.fm API was not positive. Based on Val's proposition, Kevin have looked over Spotify API functions. Through the investigation, Kevin tested with the test set up with "Get an Artist's Related Artists" to get the track info as well as testing playlist.track and track.Snapshot. The track.Snapshot code can be found in separate Google doc, "spotify_test_Kevin" in the shared Google drive. In addition, Kevin found that the Spotify has their own recommendation and visualization service, "An Artist Explorer" (https://artistexplorer.spotify.com/). Kevin checked the code of this service on Github (https://github.com/fsahin/artist-explorer) to see any possible reference to our project. After finishing observation research with the Spotify, Kevin did the basic test with code set from the

"Then Listen To" open source project (https://github.com/chrisbenincasa/thenlistento).

Kevin then spent most of his last time to prepare for the final report in the end of the project.

2.3 Yuxiang Peng (yp344)

Yuxiang found the Then Listen To and Music Recommendations that were used as a reference for our project. He also suggested an example of the D3 force layout that uses a tree structure for nodes and their edges.

2.4 Wodan Zhou (wz262)

(N/A)

3. Data Description - Spotify

Spotify is a popular online music streaming service with over 20 million songs. This paper discusses the Spotify technology and the possibilities to integrate this technology into applications. Topics such as the metadata API, Apps and the Spotify URI scheme are covered. Furthermore, the strengths and weaknesses of the Spotify technology are discussed, as well as intended and surprising applications. Finally, a 'getting started' guide is provided, with a basic explanation of using the metadata API in an application. Spotify is an online music streaming service which was created in Stockholm, Sweden, and launched in 2008. With a database of more than 20 million songs, with more than 20,000 songs added per day, Spotify is one of the

biggest music streaming services around, available in 56 countries. Creating a Spotify account allows users to create playlists and subsequently share these playlists with other users and collaboratively edit them. The sharing of these playlists is supported by many websites, which link to the playlist in Spotify. Users with an account can also integrate it with their Facebook and twitter accounts which allows them to share tracks and playlists or access those of their friends.

3.1 Data Details

The variables/endpoints of interests in the *Spotify Meta Data set* are as follows:

(1) artist

`https://api.spotify.com/v1/artists/{id}`

(2) artist's top tracks

`https://api.spotify.com/v1/artists/{id}/top-tracks`

(3) related artists

`https://api.spotify.com/v1/artists/{id}/related-artists`

(4) track (song) popularity

(5) name of track

(6) artists (on the track)

3.2 Pre-Processing – API

The AJAX calls return json objects which we can scan to find the relevant features needed.

3.3 Data Input

Because we can't return tne input data directly from asynchronous AJAX requests, we nested each request where results for one were dependent on the results from another. Once the final

request is complete we call the showMap function which populates the graph with nodes.

4. Mapping Data to Visual Elements

We decided to put all the components into our interactive design set up with one simple screen so that the users could see and understand all the functionality intuitively. Based on our preliminary storyboard, the overall layout of the visualization page includes: 1) The main system with the initial input data field, 2) Recommended music list 3) with interactive radius and lines and 4) Details of the songs we recommended. The mapping of the recommendation data is used through the linking structure that the force layout. This shows relationships between user selected songs and suggested songs.

4.1 Capstone Design Overview

Capstone Design applies the engineering sciences to the design of a system, component or process. The projects reflect practical industrial and mechanical engineering design projects or may involve a combination of both disciplines. Students choose the particular design project with approval of appropriate faculty. Design teams are organized. Each project includes the use of open-ended problems, development and use of design methodology, formulation of design problem statements and specification, consideration of alternative solutions, feasibility consideration and detailed system descriptions. It also includes realistic constraints (such as economic factors and social impact). For our project in particular we examine how users can capitalize on data visualization to meet their needs. This comes with challenges as we attempt to design a dynamic system that changes based on how the user behavior changes.

4.2 Interaction Design Considerations

Given that all the visual components should be seen in one screen, As an inspiration we looked at the simplistic interactions of the website http://www.palettable.io.

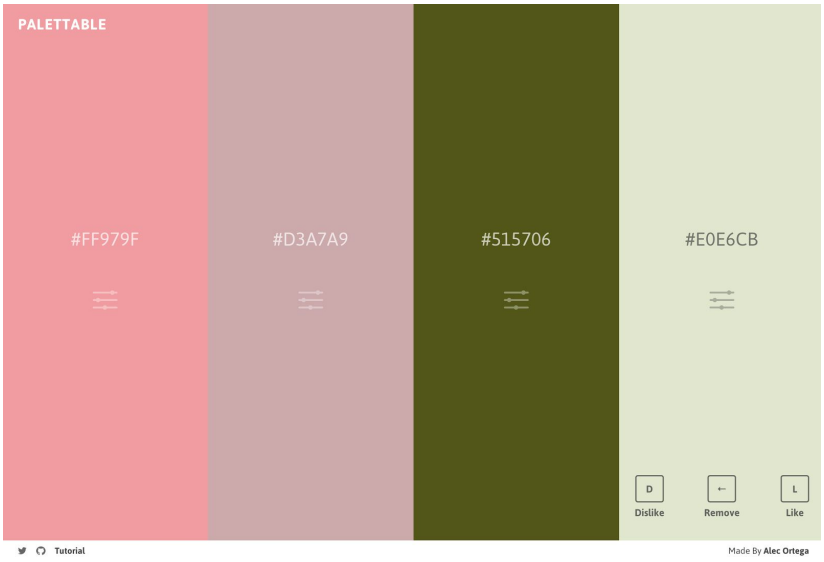


Figure 7:Example of suggested color combinations on Palettable.io

5. The Story

The visualization not only shows the overall relationships between songs on Spotify, but also the decision making behavior of the user. As they select new songs and disregard others, the visualization shows their thought process and preferences along the way, making a unique story about what songs they like or what type of mix tape they would like to make.

5.1 Does This System Really Work?

After doing numerous tests we found that our system does work. We did some initial user testing where we asked two Cornell students to use the website to create their own playlist. The results

were positive, as they felt the system was “cool” and potential something they see themselves using. They mentioned they would like to be able to see how their choices compared to that of other people and also be able to share their mixtape with their friends. These are features we can think about for a future iteration.

5.2 Extensions

Since our project mainly used a collaborative filtering-based recommender systems for our recommendation process, we might want to compare this project with possible incorporation of context-based information about the music consumption into the recommendation process with building and analyzing contextual clusters. Since there is a lot of possibility for improvements by adapting this general approach to better fit to the music recommendations problem, building a hybrid recommendation system with collaborative filters and context information as well as integrating playlist into the recommendation process for improving recommendations would be interesting topic. Therefore, the main contribution of this project would be learning about interactive design with following Capstone designing process with fun facts that the music recommendation method can be actually implemented with the new design possibility.

6. Discussion and Thoughts

This project was complex because we had to figure out how to show ever changing data in a way that was easy for users to follow and would be meaningful to them. We believe we were successful but could have made it better. We imagine this tool could be used for DJs to create playlists on the fly and would like to do more user testing to validate or invalidate this

assumption. Overall working on this project was most difficult because our team had very poor communication. For the most part, each individual on our 4 person team assumed that the other team members were making progress when they truly were not. Ultimately, one team member did 90% of the work on the website, and a different team member did 90% of the work for the final report. The others were non-responsive and not truly willing to put in the work required to complete the project. Therefore, there was a lot lacking from this project that could have made it much better. This team could have worked together much better if members had stuck to a regular meeting schedule, responded to messages in a timely manner, and reached out to the TA for help. However, the contributing team members still feel that we accomplished a strong result that could definitely be carried forward and would be a valuable tool. From this project, we hope we can learn not only about technical skills but also team leaderships we may need for the future.

We sincerely thank you to Dr. Mimno to lead this great class and we also appreciate all the support from the teaching staffs of the class.

Thank you so much.

7. References

Marin, J. A., & Armstrong Jr, J. E. (1997). Ferreting Out Capstone Design Issues in Large Scale, Complex Problems using the House of Quality. age, 2, 1.

Marin, J. A., Armstrong Jr, J. E., & Kays, J. L. (1999). Elements of an optimal capstone design experience. Journal of Engineering Education, 88(1), 19.

Kreitz, G., & Niemela, F. (2010, August). Spotify--large scale, low latency, P2P music-on-demand streaming. In Peer-to-Peer Computing (P2P), 2010 IEEE Tenth International Conference on (pp. 1-10). IEEE.

Bateira, J. L. (2014). Spotify-ed-Music recommendation and discovery in Spotify.

Lin, N., Tsai, P. C., Chen, Y. A., & Chen, H. H. (2014, September). Music recommendation based on artist novelty and similarity. In MMSP (pp. 1-6).

Hirschmeier, S., Döppner, D. A., & Schoder, D. (2015). Stating and Discussing Challenges of Radio Recommender Systems in Contrast to Music Recommendation. In DMRS (pp. 29-32).

Pichl, M., Zangerle, E., & Specht, G. (2015, November). Towards a Context-Aware Music Recommendation Approach: What is Hidden in the Playlist Name?. In Data Mining Workshop (ICDMW), 2015 IEEE International Conference on (pp. 1360-1365). IEEE.