Leveraging Networks for Artist Recommendations and Friend Insights

SI 671: Data Mining Final Project

December 13, 2021

Lauren Lunsford
M.Sc. in Information Student
University of Michigan, Ann Arbor
llluns@umich.edu

Abstract

Consumer analytics within the music industry are increasingly integral to platform success. As our data-driven world evolves, consumers will want more robust analytics. Using bespoke music streaming data, this paper establishes methods of comparing friend's music streaming behaviors. From the same data, networks representing users' personal beliefs on artist similarities are created. These networks are then leveraged to produce artist recommendations, based on a user's friend's music streaming behavior.

1 Introduction

Product differentiation among audio streaming services has become invaluable. Streaming services utilize techniques such as personalized playlists, music recommendations and streaming analytics. For instance, Spotify's *Wrapped* is a marketing technique that generated a 21% increase in application downloads within its first month in 2017[1]. It generated this publicity by providing users with insights about their own listening habits that they could learn from and share with friends. Insights such as most listened to album, song, artist, number of countries' songs a user has listened to, which artists they were in the top 1% of listeners to, etc. were reported back to users. In an effort to mimic Spotify's success, iTunes launched a similar campaign in 2020.

Spotify has also differentiated itself by its unique music recommendation algorithm. It's algorithm takes into account multiple factors including a user's individual streaming data, and aggregate global streaming data2 (SOURCE).

Listening behavior analytics and recommendations are becoming the status quo in music streaming. However, current differentiation efforts by Spotify and iTunes do not focus enough on user’s friendships. User analytics marketing campaigns revolve around individualized insights with simple statistical analyses that focus on individual users. Music recommendation software uses personal streaming data and may focus on aggregate streaming data, but does not publicly provide the ability for friends to automatically provide music recommendations. As audio streaming services work harder to differentiate themselves, utilizing the social aspect of music to report back shared analytics and personalized recommendations will be advantageous.

2 Problem Definition

Listening analytics campaigns have utilized similar insights for years. As users become more accustomed to personalized insights, new methods of analysis will be required to maintain attention and intrigue. As we are increasingly living in a data driven world, interest in metrics that can describe qualities of friendships, such as music taste, will be in increasing demand. This work builds an approach where two people can quickly compare their music similarity score by data analytics, and use that data to quickly receive artist recommendations curated by their friends without effort.

This paper establishes a methodology to utilize network analytics to produce artist recommendations. The pairwise analytics utilizes minutes listened to common artists to establish a simple pairwise comparison of friends. It then uses the shared artists as a fine-tuned dataset from which to ask for artist recommendations from friends’ artist networks. An artist network is a network where artists are linked if two artists are played within the same day for more than 4% of days collected.

3 Data

The data is a collection of four users complete Spotify streaming data. The data collection was performed by the author and includes the author’s personal data. Of the four users, the author is identified as the primary user, and the three additional users are the primary user’s friends. A network visualization of the relationships of the users has been provided in Figure 1.

Relationships of Users

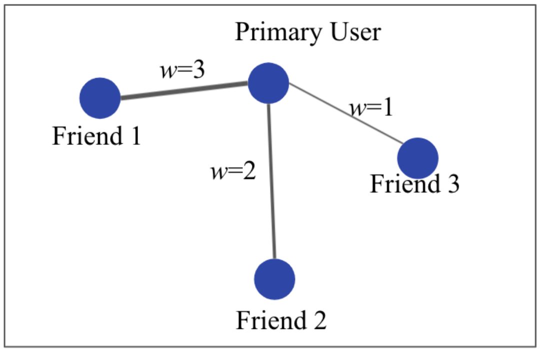


Figure 1: A visual representation of users' friendships. The weight signifies the closeness of the friendship where 3 is the most close and 1 is the least close friendship.

Each user requested their personal data from Spotify and was emailed a zip file containing multiple files, including their streaming data. Each user's streaming data includes each instance of a song or podcast listened to for a year. Each instance includes the song name, artist name, album name, length the audio item was listened to, and time that the audio item was ended. There are approximately 7,000 to 50,000 instances per user.

4 Related Work

Prior approaches to analyzing music listening data have mirrored the approaches of Spotify's wrapped. Yizhack creates a dashboard that visualizes top songs, top languages of songs listened to, and favorite artists[3]. Buentello also looks at individual statistics, regarding top artists, songs, which days of the way a user listens to music, and a quick feature analysis[4]. These approaches seek to provide insight on individual user listening behavior through rudimentary statistics such as top artists, top songs and time listened by day of week.

Contrary to those approaches is Zhang et al. The paper analyzes arrival patterns, playback arrival patterns, and daily variation of session length as well as favorite listening times of day and the downtime of successive user sessions[5]. It varies from individual user data to provide a more holistic understanding of user listening data. However, this paper also does not provide comparisons of multiple users' audio data.

Industry knowledge regarding music recommendation algorithms are considered proprietary and are often not shared. A 2015 copy of Spotify's *Discover Weekly* algorithmic strategy was the most recent insight available. The algorithm uses two main approaches in producing music recommendations: exploit and explore[6]. The algorithm utilizes existing knowledge of a user to *exploit*, or give the user more of what they like. It then *explores* by taking worldwide aggregate data to find new music for the user. This aggregated data integrates the use of additional people to make recommendations but does not take advantage of the smaller subsets of friendships users might share.

5 Methodology

Preprocessing

Data cleaning consisted of 3 primary steps: changing time zones, converting milliseconds to minutes, and dropping rows where listening time was less than 30 seconds.

When plotting song frequency by hour, the distribution appeared shifted from self-reported listening times by users. To ensure accurate time data, time was localized from UTC time to ET time using the `pytz` package as all users were in the eastern time zone. The original and shifted listening distribution can be seen in Figure 2.

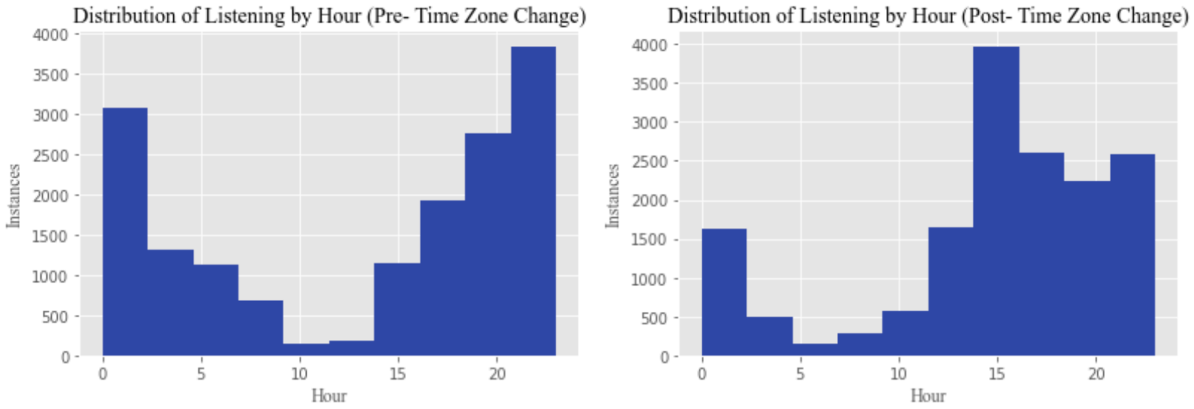


Figure 2: Frequency of audio instances by hour. The plot on the left plots the distribution before preprocessing, when the data was in UTC time zone. The plot on the right shows the distribution after preprocessing, when the data was localized to the ET time zone.

The variable indicating the length an audio tiem was listened to, listening time, was initially in milliseconds. This variable was converted to minutes for easier comprehension and comparison.

Finally, instances where the audio item was listened to for less than 30 seconds were dropped. When plotting frequency of audio items against length of time listened, there was a large density of instances less than 30 seconds as seen in the left plot of Figure 3. Anecdotally, this appeared to be because of songs that were skipped after listening to a few seconds. Skipped songs indicate a user did not want to listen to that song, and that it did not fit into the user's mood. Streaming data will be utilized to create artist itemsets, grouped by day. These itemsets will build networks where similar artists are connected if they meet the minimum support threshold of 4%. Artist networks represent the similarity of artists, as grouped by the individual listener. This relies on the assumption that artists/songs listened to on the same day were actively selected together. Actively selected artists/songs indicate a decision to group by mood. Without active selection, this assumption cannot be true. Unwanted and skipped songs are not actively selected for. Therefore, unwanted songs are not helpful data, therefore songs under 30 seconds were dropped from the final dataset. The cleaning resulted in a 30.8% reduction of datapoints, but only a 1.0% reduction of total minutes listened. The new distribution can be seen on the right plot of Figure 3.

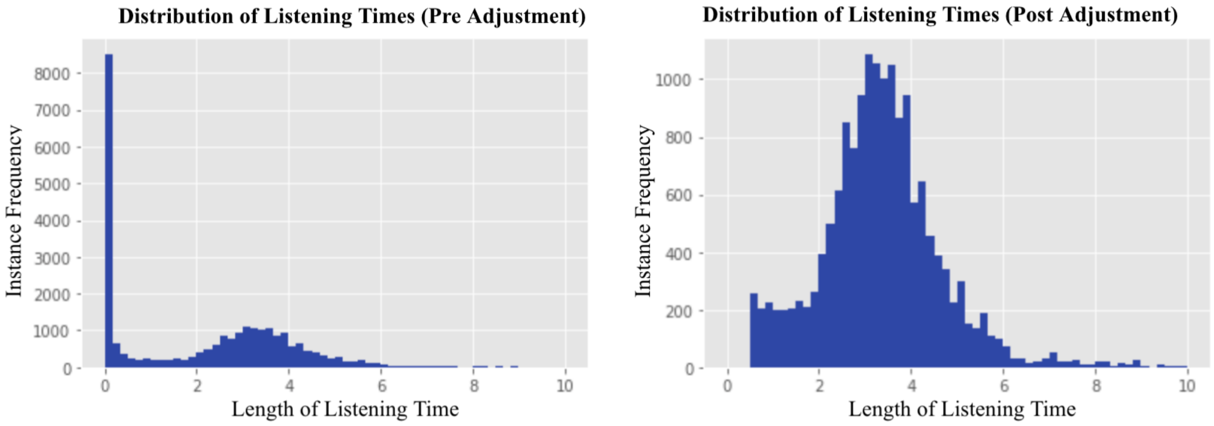


Figure 3: Distribution of listening times by frequency. The large spike in the left plot visualizes the dense number of songs skipped or partially listened to. The right plot visualizes the distribution after instances less than 30 seconds were dropped from the data.

Individual Itemset Similarity Analysis

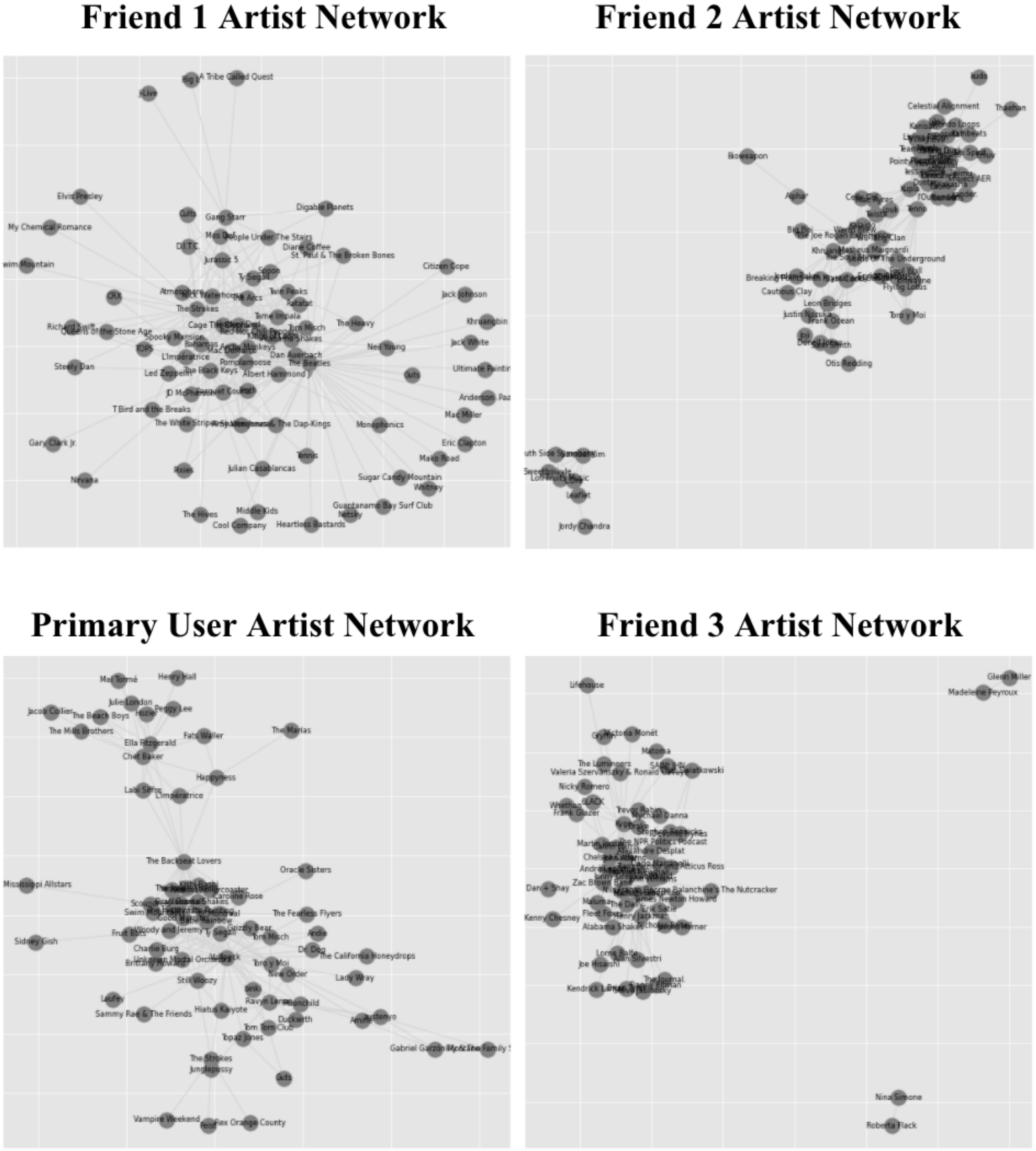
The cleaned data was then aggregated into itemsets by artist by day. For instance, if a user listened to The Beatles, Men I Trust, and Beyonce on the same day, the itemset would be: `{‘The Beatles’, ‘Men I Trust’, ‘Beyonce’}`.

Artists, not tracks/songs, were chosen to increase the available data. While individual tracks/songs may provide more specific insight regarding similarity, using artists allowed for more robust data, and a higher support. For instance, if two itemsets were `{‘Single Ladies-Beyonce’, ‘Octopus Garden- The Beatles’}` and `{‘Formation- Beyonce’, ‘Eleanor Rigby-The Beatles’}`, These two itemsets would not have any similarities. However, if they were `{‘Beyonce’, ‘The Beatles’}` and `{‘Beyonce’, ‘The Beatles’}`, they would be considered similar and provide us with more information. Itemsets were grouped by day as no standard grouping of listening exists.

For each user, itemset similarity was performed with a support of 0.04, and a minimum k-itemsets of 2. This means that two artists must coexist in a minimum of 4% of all itemsets (days). These values were decided after numerous experiments and visualizing. This value provides enough artist network edges for value to be derived, without adding in superfluous connections.

Individual Artist Networks

Using the results of the itemset similarity, an artist network was created. 2-itemsets from the itemset similarity results were fed into an individual user’s network as edges. In this network graph, nodes are artists, and edges indicate that they shared a minimum support of 0.04. In other words, edges indicate artist's songs were played within the same day in at least 4.0% of days collected. An example of all user’s artist networks can be seen on the next page in Figure 4.



Friend 1 Artist Network

Friend 2 Artist Network

Primary User Artist Network

Friend 3 Artist Network

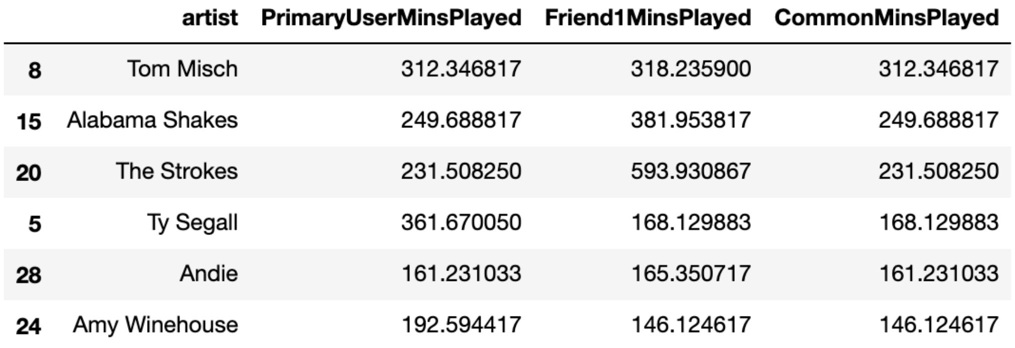
Figure 4: Artist networks for each user.

The artist network represents artist similarity. This declaration relies on a core assumption: Users curate their listening experience by skipping and selecting songs. Users select artists and songs for their specific mood on given days. If a pair of artists/songs are played within the same day, they have similar moods. If a pair of artists meet the minimum support threshold, they are therefore similar artists.

By assuming the aforementioned statements, the created networks provide a personalized representation of the user's grouping of artists. These networks provide insight into which artists a user thinks are similar, as well as the user's music consumption behavior.

Pairwise Artist Streaming Similarity

To make relevant queries for artist recommendations, users' shared listening time of artists were obtained using pandas. User's individual streaming data was grouped by artist and the listening time was summed. These grouped data frames were then merged with a friend's grouped data frame by artist to see all common artists. The minimum shared minutes per artist was calculated by taking the intersection of minutes listened. If one user listened to The Beatles for 200 minutes, and the other listened for 240 minutes, their shared time listening to The Beatles was 200. This method of calculating shared interest allowed for the sorting of artists based on mutual preference. An example of a shared artist streaming comparison can be seen in Figure 5.



	artist	PrimaryUserMinsPlayed	Friend1MinsPlayed	CommonMinsPlayed
8	Tom Misch	312.346817	318.235900	312.346817
15	Alabama Shakes	249.688817	381.953817	249.688817
20	The Strokes	231.508250	593.930867	231.508250
5	Ty Segall	361.670050	168.129883	168.129883
28	Andie	161.231033	165.350717	161.231033
24	Amy Winehouse	192.594417	146.124617	146.124617

Figure 5: Minutes played of common artists among the primary user and friend 1. This dataset displays only the top 6 results.

Taking an average of the two times would have led to unbalanced artist interests. If the two prior users had listened to The Beatles for 20 and 400 minutes respectively, the average listening time would have been 210 minutes. This would have ranked The Beatles as a high priority artist for the first user, even though they only listened to the artist for 5.0% of the time the second user listened.

Artist Recommendations

Artist recommendations are taken from the individual user artist networks. Operating under the aforementioned assumption, edges in the artist network indicate similarity. Artist recommendations queries are derived from the artist streaming similarity data frame. For each shared artist, starting from the mutually most streamed artist, recommendations are provided. Artist recommendations are taken from the primary user's friend's network. User A would look to user B's network. Recommendations are the network neighbors of the artist query.

6 Evaluation and Results

Evaluation Metrics

Evaluation of artist recommendations was not possible within the scope of the project. To properly evaluate, data would need to be collected post-artist recommendation to see if users adopted artist recommendations and increased their consumption. Due to the time limitations, this could not be performed.

Artist Streaming Similarity

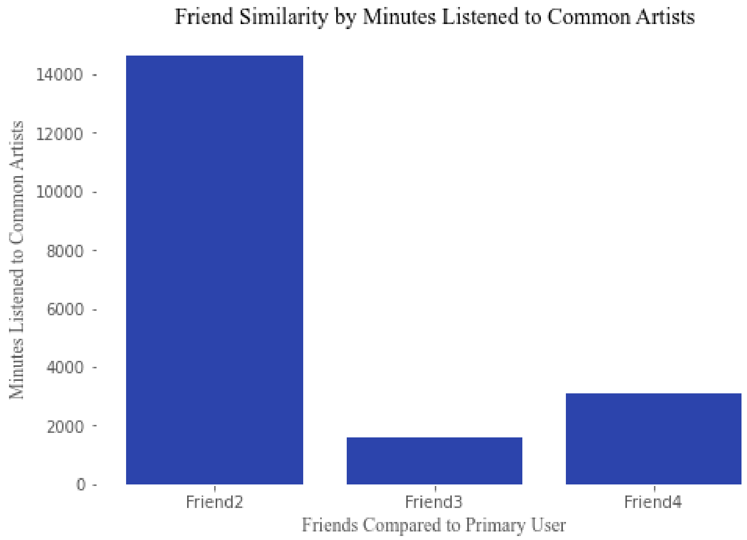
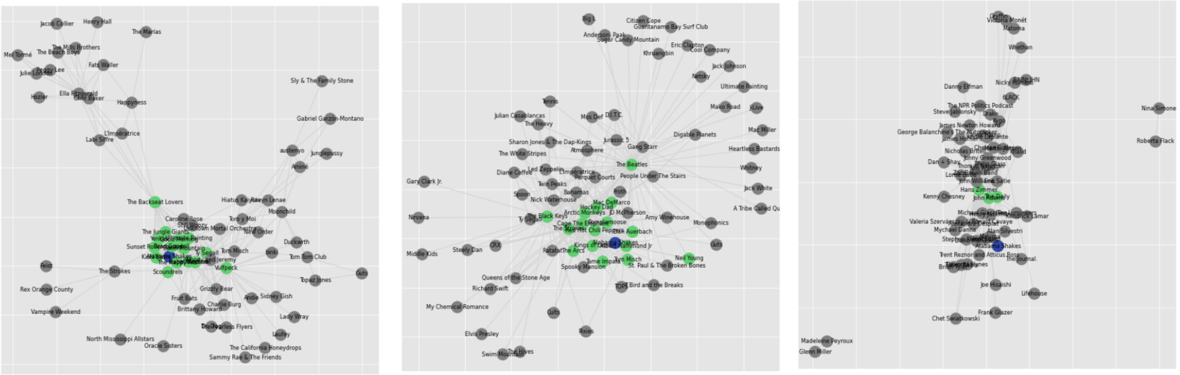


Figure 6: Total minutes played of common artists between each friend and the Primary User.

A visual representation of total minutes of shared artist streaming can be seen in Figure 6. As no validation metric can be applied, for reference, the bar plots are ordered by self-reported friendship levels. While I do not make the argument that friendship closeness is indicative of highly similar music listening habits, the bar plot shows some relationship between closeness and shared minutes. This number can be used as a quick gauge for music similarity. In conjunction with top shared artists in Figure 5, this metric provides a quick snapshot into what friends have in common. This can facilitate further conversation over commonalities or encourage music sharing[1*].

Artist Recommendations


Recommended Artists Similar to 'Alabama Shakes' Across Multiple User's Artist Networks

Primary User Artist Network

Friend 1 Artist Network

Friend 3 Artist Network

Recommendation:

```
If you like Alabama Shakes, you might
enjoy: ['Babe Rainbow', 'Brad Goodall',
'Good Morning', 'Kishi Bashi',
'Scoundrels', 'Sunset Rollercoaster',
'Swim Mountain', 'The Backseat Lovers',
'The Happy Fits', 'The Jungle Giants',
'Ty Segall', 'Ultimate Painting',
'Vulfpeck', 'Yenkee', 'of Montreal']
```

Recommendation:

```
If you like Alabama Shakes, you might
enjoy: ['Albert Hammond Jr', 'Arctic
Monkeys', 'Cage The Elephant', 'Dan
Auerbach', 'Hockey Dad', 'Kings of
Leon', 'Mac DeMarco', 'Neil Young',
'Pomplamoose', 'Red Hot Chili
Peppers', 'Tame Impala', 'The Arcs',
'The Beatles', 'The Black Keys', 'The
Strokes', 'Tom Misch']
```

Recommendation:

```
If you like Alabama Shakes, you might
enjoy: ['Hans Zimmer', 'John Adams',
'The Daily']
```

Figure 7: The graphs are visual representations of a user's artist graph. The blue node in each graph is the query node. This is the artist that the primary user wishes to find recommendations for. In this example, the artist "Alabama Shakes" is queried in each user's network. The green nodes are the network neighbors of the query node, which are returned as recommendations. Underneath each graph is the resulting recommendation for the same query, but for each unique artist graph. Friend 2 was not included as they did not have Alabama Shakes in their network.

Sample recommendations can be seen in Figure 7. The figure above details how artist recommendations are bespoke to each user. In the figure, the same query artist, 'Alabama Shakes' was used to derive completely different artist recommendations. The novelty of using individual user's artist networks as the basis for recommendation, is that your query results are dependent on the taste of that given user. If a user really respects their friend's music taste from what they've shared, and wants artists similar to Alabama Shakes, they can check out who that friend would recommend without having to ask their friend.

[1*] Anecdotally, the author immediately sent these results to Friend 1 as a way of validating their level of friendship. Despite the full awareness that friends can have dissimilar music tastes, the author reported that by sharing that metric, they felt they had gained additional social capital with this Friend 1.

7 Discussion

Limitations

The established methodology relies heavily on the aforementioned assumption that the artist network indicates artist similarity as described by said user. There are no current metrics to confirm this, which is a large limitation to this paper.

Furthermore, the paper sets a minimum support value of 0.04 based on the visual appeal of the resulting artist networks. This value cannot be justified empirically. Higher minimum support values may result in sparse artist networks, but may more accurately represent similarity among artists. Inversely, lower minimum support values may result in more populous networks, but the lower threshold may decrease the reliability of similarity among artists.

Future Studies

Future iterations of this project include identifying clusters of listening smaller than days for itemset collection and comparing artist vectors for users and friends.

Item sets were grouped by day as these were the most convenient units of listening available to group by. Future studies would seek to establish an algorithm to identify clusters of listening within time. One approach to do so might include using large time gaps between listening as boundaries for itemsets. Establishing more succinct itemset boundaries would increase the likelihood that artist networks represented artist similarity.

Future studies may also utilize streaming data to create vectors of artists from individual user's streaming data and compare the vectors of shared artists. This would establish another metric of determining similarity among friends' music streaming habits.

Lastly, the creation of a metric to validate artist similarity networks would be prudent. Possible metrics include asking users to cluster given artists together based on perceived similarity.

Acknowledgments

This paper could not exist without the friends who graciously volunteered their data up to my scrutiny. Thank you.

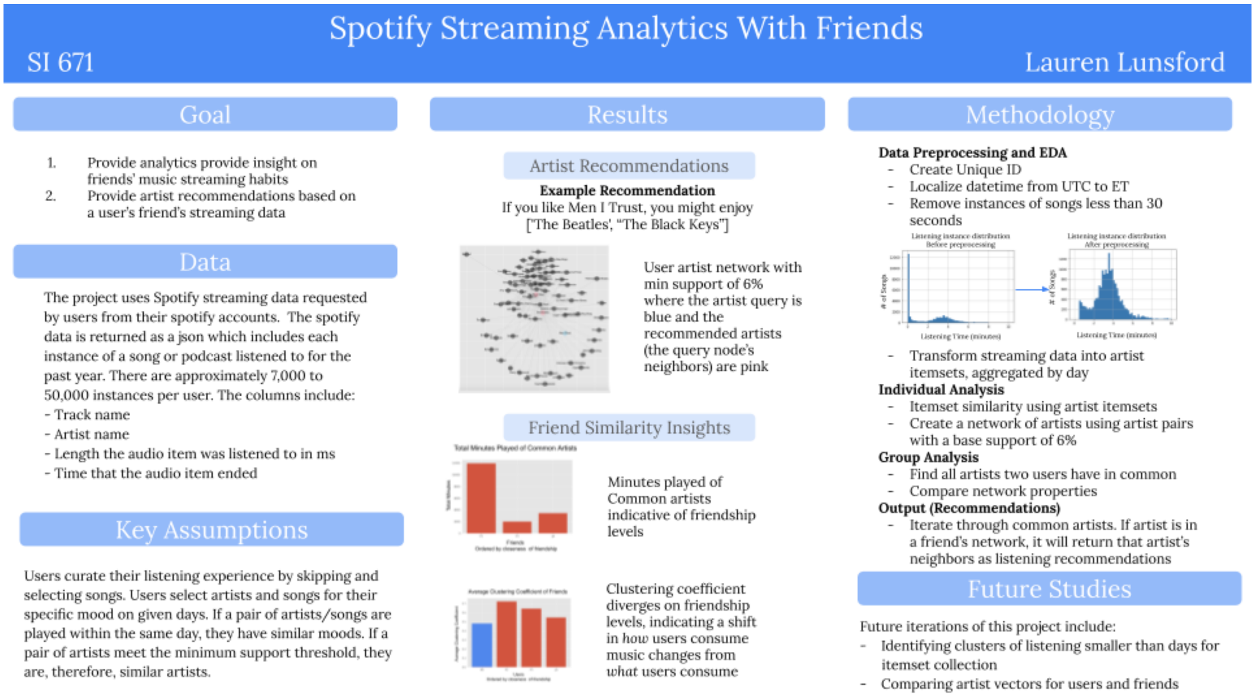
References

1. Jain, Pulkit. "How Spotify Wrapped 2020 Marketing Campaign Boosted Mobile App Downloads and Engagement." *MoEngage Blog*, 27 June 2021, https://www.moengage.com/blog/spotify-wrapped-2020-app-downloads-engagement/.
2. Johnson Follow Engineering Manager - Recommendations and Personalization, Chris, and Edward Newett. "From Idea to Execution: Spotifys Discover Weekly." SlideShare, 16 Nov. 2015, https://www.slideshare.net/MrChrisJohnson/from-idea-to-execution-spotifys-discover-weekly/31-1_0_0_0_1.

3. Buentello, Saúl. “Explore Your Activity on Spotify with R and ‘Spotifyr’: How to Analyze and Visualize Your Stream...” Medium, Towards Data Science, 6 Feb. 2021, https://towardsdatascience.com/explore-your-activity-on-spotify-with-r-and-spotifyr-how-to-analyze-and-visualize-your-stream-dee41cb63526.
4. Yitzhak, Yaron. “A Simple Guide to Visualizing Your Spotify Listening Data... Badass-Ly.” TNW | Tech, 27 Apr. 2021, https://thenextweb.com/news/a-simple-guide-to-visualising-your-spotify-listening-data-badass-ly
5. Zhang, Boxun, et al. “Understanding User Behavior in Spotify.” 2013 Proceedings IEEE INFOCOM, 2013, https://doi.org/10.1109/infcom.2013.6566767.
6. Gershgorn, Dave. “How Spotify's Algorithm Knows Exactly What You Want to Listen To.” *Medium*, OneZero, 4 Oct. 2019, https://onezero.medium.com/how-spotifys-algorithm-knows-exactly-what-you-want-to-listen-to-4b6991462c5c.

Supplemental Materials

A. The poster presented at the December 10th Exposition. Due to a re-scope of the project, a new poster was uploaded to the front of this report.


Total Minutes Played of Common Artists
Total Minutes
Friends
Ordered by closeness of friendship
Average Clustering Coefficient of Friends
Average Clustering Coefficient
Users
Ordered by closeness of friendship
Listening instance distribution
Before preprocessing
# of Songs
Listening Time (minutes)
Listening instance distribution
After preprocessing
# of Songs
Listening Time (minutes)

Spotify Streaming Analytics With Friends

SI 671

Lauren Lunsford

Goal

1. Provide analytics provide insight on friends' music streaming habits
2. Provide artist recommendations based on a user's friend's streaming data

Data

The project uses Spotify streaming data requested by users from their spotify accounts. The spotify data is returned as a json which includes each instance of a song or podcast listened to for the past year. There are approximately 7,000 to 50,000 instances per user. The columns include:
- Track name
- Artist name
- Length the audio item was listened to in ms
- Time that the audio item ended

Key Assumptions

Users curate their listening experience by skipping and selecting songs. Users select artists and songs for their specific mood on given days. If a pair of artists/songs are played within the same day, they have similar moods. If a pair of artists meet the minimum support threshold, they are, therefore, similar artists.

Results

Artist Recommendations

Example Recommendation
If you like Men I Trust, you might enjoy
['The Beatles', “The Black Keys”]

User artist network with min support of 6% where the artist query is blue and the recommended artists (the query node’s neighbors) are pink

Friend Similarity Insights

Minutes played of Common artists indicative of friendship levels

Clustering coefficient diverges on friendship levels, indicating a shift in *how* users consume music changes from *what* users consume

Methodology

Data Preprocessing and EDA
- Create Unique ID
- Localize datetime from UTC to ET
- Remove instances of songs less than 30 seconds
- Transform streaming data into artist itemsets, aggregated by day

Individual Analysis
- Itemset similarity using artist itemsets
- Create a network of artists using artist pairs with a base support of 6%

Group Analysis
- Find all artists two users have in common
- Compare network properties

Output (Recommendations)
- Iterate through common artists. If artist is in a friend’s network, it will return that artist’s neighbors as listening recommendations

Future Studies

Future iterations of this project include:
- Identifying clusters of listening smaller than days for itemset collection
- Comparing artist vectors for users and friends