Unveiling the Music Graphs: Analyzing Spotify's Users and Artists

Cristiana Lazar[a], **Christian Bakos Leirvåg**[a], **and Wee Yang Sim**[a, b]

[a]Technical University of Denmark; [b]National University of Singapore

This manuscript was compiled on December 12, 2022

We aim to empirically investigate, between listeners and musicians alike, whether music can bring individuals together and foster a sense of solidarity among those with similar musical preferences. To test this hypothesis, we first processed information on Spotify listening trends from a plethora of different sources (Kaggle, Spotify and Genius APIs). Then, we conducted extensive network and sentiment analysis of communities within Spotify networks. We show that different individuals within the same communities do indeed exude promising similarities in the realms of sentiment analysis and network properties. We found that the sentiment scores of the five most frequent genres appear similar. Furthermore, we discovered that artists and users are grouped according to their similarities. In the case of the artists' network, they collaborate or are related to artists from similar music genres. For the users, they gather with other users who share similar songs and music genres. Moreover, the most popular artists who are listened to by a community of users form communities within the artist networks. Our paper is expected to serve as a starting point for understanding users' and artists' behaviours. As technology and music intertwine and as more music gets distributed to consumers, the knowledge of our musical preferences will be beneficial and relevant for music recommendations.

social graphs| music industry | music streaming platforms| graph analysis | text analysis | sentiment analysis

As a constant in our lives, music has evolved rapidly in recent years. From a technological perspective, new genres have emerged and it's now much more accessible for listeners to discover new artists and songs. Listeners inherently create communities and form social interactions around them.

In this paper, our purpose is firstly to analyse the music industry by looking at how artists interact with each other, how they impact users, and how certain user groups interact and cluster together. Secondly, we want to analyse the sentiment of the lyrics, how that sentiment influences specific genres, and how specific words impact a genre.

Our musical behaviour can be easily depicted once we make the music we are listening to public. To define their behavior, a user's confidential data need not be compromised. Twitter facilitates posting the music a Spotify user is currently listening to using the hashtags #nowplaying, #listento or #listeningto. A group of researchers explored this opportunity by creating a dataset where pieces of information about the tracks a user is listening to have been gathered(1). We are particularly interested in this dataset since it is a sample of real Spotify users and allows us to verify our hypotheses regarding the behaviour of users and artists in the music industry. The dataset offered only the tracks and artists' names. Since we were also interested in text analysis, we gathered the tracks' lyrics from Genius.

Our findings rely on studying the artists' and users' networks. We studied their characteristics using graph analysis tools and tested a series of hypotheses created based on our understanding of the real world and their expectations. All of these will be discussed in the first part of the Results. Furthermore, findings about the impact of the tracks' lyrics will be presented in the second part during the text analysis discussion.

Results

Our study relies on real user data gathered from Twitter, which comes with advantages and disadvantages. Firstly, we expect the data to be representative of the real users' distribution since it has been sampled from Twitter. This brings an additional layer of complexity as it inherently captures the human social groups. That will result in more representative user communities in the users' network. However, the drawback is that the dataset is predisposed to human error and requires preprocessing. For instance, in the case of tracks performed by collaborating artists, there was no consistent means used to separate the artists' names. As a result, users have chosen to separate them in a myriad of different ways, which made our preprocessing task difficult and susceptible to outliers.

Data Sampling. During the preprocessing, we discovered a fascinating phenomenon; the number of unique artists in the music industry has exceeded two hundred thousand. We attributed the high number of artists to the fact that it is now easier than ever to release new songs. Anyone can record and release a track without any intention of it being popular. With that in mind, we reduced the number of artists by sampling the most popular ones. These artists will significantly influence

Significance Statement

The results of this study will be beneficial to society since music is part of everyone's daily ritual. The need for song recommendations continues to grow as technology and music become more intertwined and more music is distributed to the users. Users who adopt the suggested strategies from the study's findings will therefore have a better understanding of their own musical preferences. Users that share the music they are currently listening to will be guided on which artist, genres and moods they most empathise with. Artists will be able to see how they are connected to one another based on their popularity amongst the users, how they collaborate, and how similar they are.

Author contributions

[1] Cristiana Lazar contributed to data cleaning and preprocessing, network generation, users' network analysis, and text analysis. Christian Bakos Leirvåg contributed to Spotify artist's network analysis. Wee Yang Sim contributed to the collaboration artist's network analysis and text analysis.

[2] To whom correspondence should be addressed. E-mail: s222698@student.dtu.dk

the user network structure. Since the artist's popularity has a tail-heavy distribution, we chose our sampling threshold as the mean value of the artist's popularity, reducing the number of artists to 22025. Furthermore, we extracted the largest connected component to remove the non-influential artists from the network.

Due to computational limitations, we had to sample a subset of users. The initial number of 15267 users would have resulted in a very dense network that would be difficult to process since the dataset contains users listening to over two hundred thousand tracks. Therefore, we decide to keep around 90% of the users by removing the ones who listen to over 600 tracks. This decision has been made based on the assumption that if a user is listening to large numbers of various tracks, they cannot provide meaningful contributions to the community they are part of, and they might end up as link nodes between communities.

We base our findings on three networks. The first network is an artists' network based on artists' collaborations. We wanted to assess whether the sample the artists' collaboration network was constructed with was truly reflective of the real-world distribution. We then constructed a second artists network based on Spotify's related artists. The third network is the users' network, where the connections are made if users are listening to the same tracks.

Spotify Artist Network #1 (Links via collaboration).

Nodes and edges of the network. Each node represents a Spotify artist from the #nowplaying dataset. If there is a collaboration between two artists, we include an edge between the two nodes representing the artists. Hence, the network contains *4238* artists (nodes) and *9067* instances of artist collaborations (edges).

Density of the network. We observe that the density of the graph is very low at *0.1%*, which tells us that artists don't necessarily collaborate as much as they could. This makes sense as an artist is likely to be selective with picking their collaborators as not all genres are able to mesh well with each other. The selective and intentional nature of collaboration would hence lead to a low artist network graph density.

Degree Distribution. Given the context, the degree of the nodes represents the number of collaborations an artist has with other artists. By examining the degree distribution (Figure 1a), we observe that most artists have few collaborations, around two, and a few exist with many collaborations, which might denote the presence of hubs. The artist's degrees also have a weak positive correlation with an artist's popularity. We cannot definitively determine whether one causes another, but one possible conclusion is that an artist that collaborates with other artists more (and has a higher degree) will have a higher popularity amongst fans as their songs would reach out to wider audiences.

The artist network's degree distribution obeys a power law. In other words, the artist network is a scale-free network. The power law coefficient we obtained from our analysis is about *3.60*. Hence, the artist network follows the small world rules since the degree exponent γ is greater than 3. Inherently, this tells us that the spread in terms of collaborations between artists is quantifiable and does not diverge. The average distance between the artists follows the small-world result that's derived for random networks. For scale-free networks where $\gamma > 3$, hubs continue to be present (as seen from some artists that have many collaborations). However, they are not sufficiently large and numerous to have a significant impact on the distance between the nodes. This means that the distances between each artist in the network are larger compared to another artist network with a smaller degree exponent γ.

Clustering Coefficient. The majority of the artists have a clustering coefficient between *0%* and *20%*. This is likely due to most of the artists having a clustering coefficient of 0. This means that the collaborators of these artists are not linked to each other at all. On the contrary, we can see that there is a sizeable chunk of artists with clustering coefficients ranging from 0.8 - 1. This means that these artists are collaborating with other artists that are also likely to collaborate with each other. Node degree and clustering coefficient appear to have no correlation with each other. This means the artists that have a lot of collaborations are collaborating with artists that do not collaborate with each other. This is unsurprising as hubs (artists with lots of collaborations) may be linked to mostly nodes with low degrees (artists with a few collaborations).

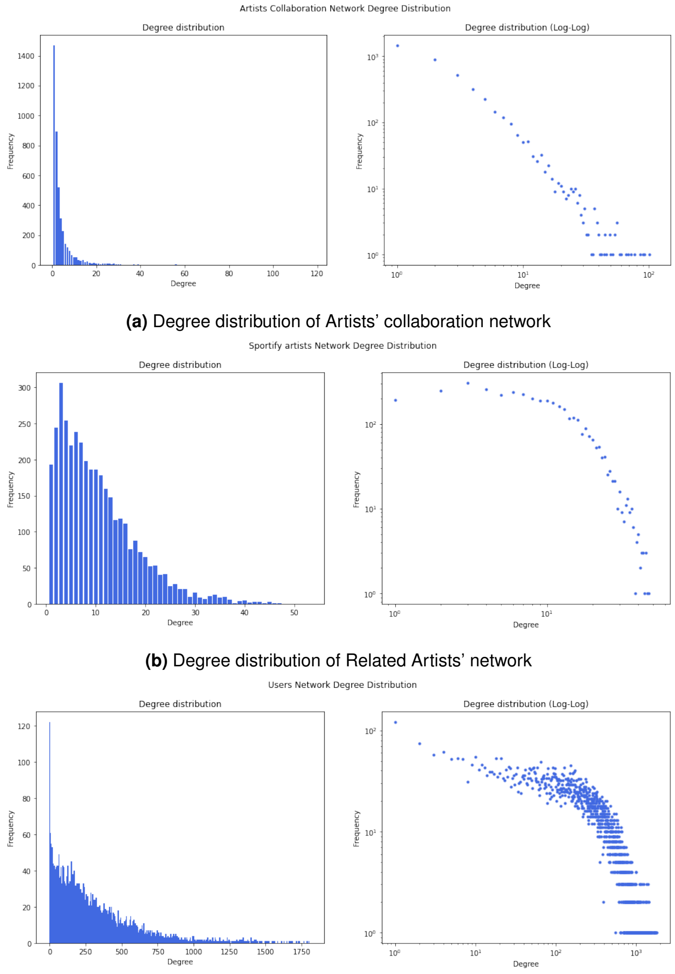
Centrality. We opt to use betweenness centrality as it determines the importance of artists based on the number of times they occur within the shortest paths between other nodes. The average betweenness centrality for the entire artist network is only *0.128%*. This means that most of the artists are not in the shortest paths between artists. This is expected given the large size of the scale-free network. Centrality is not correlated to degree distribution but has a weak positive correlation with the popularity of an artist. This is apt as a more popular artist would be more central and prominent amongst the shortest paths of other artists.

Spotify Artist Network #2 (Links via related artists).

Nodes and edges of the network. In contrast to the previous network, the *Spotify Artist Network 2* establishes links between artists based on their related artists we get from the *Spotipy API* call. By creating this network, we hope to supplement our prior understanding of the artists' communities by providing an alternative perspective. This network consists of *3778* nodes and *19524* edges.

Density of the network. One observation is that this network has significantly more edges than the network based on collaborations between the artists. Despite this, the network has a density of *0.27%*, which implies the graph is not particularly dense. By having a higher density, we can conclude that artists in the network are more related than their collaborations might suggest. There might be some underlying relations that collaborations alone do not capture.

Degree distribution. Most artists have a degree equal to 3, in contrast with the first network, where the mode value of the degree distribution was 1. This means that an artist would have more related artists than collaborations. Intuitively, this makes sense as being related is more probable than an actual collaboration. As in the artists' collaboration network, the degree distribution (Figure 1b) follows the power law, with a power degree equal to *5.62*, which will also place the network in the small world regime, like the first artists' network.



(a) Degree distribution of Artists' collaboration network

(b) Degree distribution of Related Artists' network

(c) Degree distribution of Users network

Fig. 1. The degree distribution of the networks plotted both on a linear scale (left) and a logarithmic scale (right). It is noteworthy to point out that all three networks' degree distribution follows a power-law distribution.

Clustering coefficient. The average clustering coefficient is around *40%*, telling us that the artist's neighbours are not always related to each other. This also implies that there is some diversity among the artists in the communities. When inspecting the communities of the network, we find they have a wide spectrum of different clustering coefficients, though most communities range between *40% to 60%*. There is also no correlation between the node degree and the clustering coefficient.

Centrality. The centrality of the network was, once again, measured by betweenness centrality. There is no correlation between centrality and degree distribution, which means that the number of related artists that an artist has will not influence their centrality. However, we discovered a slight positive correlation between the popularity and centrality of an artist. This means that an artist popular within the dataset tends to have a higher centrality. We see that the most central artists in the network are: *Amy Winehouse and (Will) Pharrell*, who are considered to be two fairly popular artists.

Spotify Users Network Analysis.

Nodes and edges of the network.. Each node represents a Spotify user from the #nowplaying dataset. Suppose two users are listening to the same tracks. In this case, we connect them by a link weighted with the sum of the inverse frequency of the tracks. We wanted to reduce the influence of the most popular tracks since a track that reaches a notable position on a top chart would have many confounding variables that might be misleading for our user community analysis. The users' network consists of *13,130* nodes and *1,791,938* edges.

Density of the network. A density of *2%* for our users' network's backbone indicates that there are very few connections among the users. In this case, we retain only the most meaningful connections after the backbone extraction, resulting in a more meaningful conclusion about the users' network structure and the users' interactions.

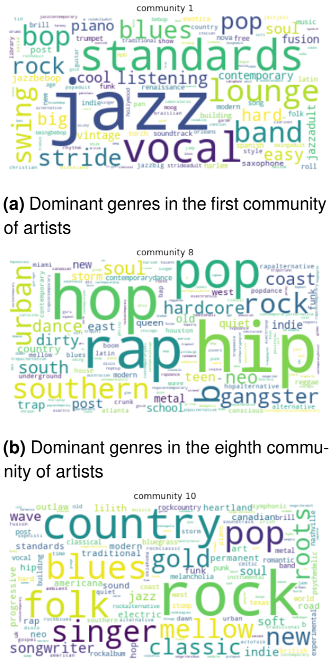
Degree distribution. In this network's case, the nodes' degree represents the number of tracks that a user has in common with other users. Analysing the degree distribution (Figure 1c), we observe that it follows the power law, with an exponent degree γ equal to *4.41*, indicating that the users' network follows the small world's rules. An interesting observation about the users' network is that the degree distribution positively correlates with the number of tracks a user listens to. This observation is not surprising since we expect a user listening to a higher number of tracks to have a higher probability of having more in common with other users.

Clustering coefficient. The average clustering coefficient has a relatively low value, only *22.5%*, which means that the neighbours of a user do not necessarily share the same tracks or, to a certain extent, the same music tastes. With this observation, we expect a higher diversity of artists and music genres in users' communities.

Centrality. As a centrality measure, we have used eigenvector centrality since we are interested in measuring the user's influence on the network by the music they are listening to. In our analysis, we discover that centrality has a strong positive correlation with the degree and also with the number of tracks a user listens to. This means that a user is more influential, the more he is connected to other users and the more songs he listens to.

A. Findings and Testing of hypotheses.

Artists with similar music genres should form communities.. We would expect artists that perform music of similar genres to form communities. Intuitively, this makes sense as artists within the same genre can collaborate on new songs easily. To investigate this, we plotted the genre word clouds of different communities within the artist network. If we could identify underlying themes in the genres of each community, it would validate our hypothesis. Our results show that each community had artists that were *distinctly* of several related genres. For example, community 10 (Figure 3c) was undoubtedly an old-school rock, country and folk community. On the other hand, community 1 3a had jazz-oriented artists, with words like 'jazz', 'standards' and 'swing' being prominent throughout the community's artist genres. As such, we can say, with

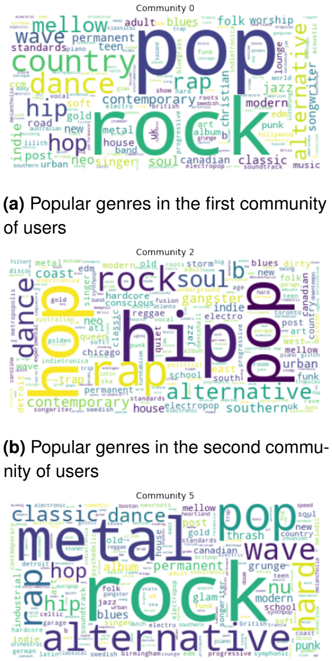


(a) Dominant genres in the first community of artists

(b) Dominant genres in the eighth community of artists

(c) Dominant genres in the tenth community of artists

Fig. 2. A subset of communities from collaboration artists' network where the dominant genres are displayed in the form of a word cloud. (a) community of jazz-oriented artists (b) community of hip-hop-oriented artists (c) community of rock-oriented artists



(a) Popular genres in the first community of users

(b) Popular genres in the second community of users

(c) Popular genres in the fifth community of users

Fig. 3. A subset of communities from users' network where the popular genres are displayed in the form of a word cloud. (a) community of pop-rock-oriented users (b) community of hip-hop-oriented users (c) community of rock-metal-oriented users

high confidence, that artists are generally grouped together according to the genres of their music.

Users with similar preferences should form communities. A common expectation for the users' network is that users with similar musical tastes cluster. We validated this hypothesis in two different ways. Firstly, we computed the average percentage of a user's tracks that are also listened to by other users in the same community. The results show that in the majority of communities, in the case of more than 50% of users, over 80% of their songs are listened to by other users from the same community. The hypothesis also was validated by computing word clouds that emphasise the most dominant genres in a user's community. Since an artist can belong to multiple genres, it was expected that an artist would appear in multiple genre word clouds. However, we can easily distinguish the difference between communities, which means that users are grouped according to musical preferences.

The user's network would be impacted by the communities from the artist's network. By creating a word cloud for the most popular artists in the user's communities, we could potentially determine whether the artists' communities affect what the users listen to. This is done by comparing the word cloud from the users' network to the word cloud with the most popular artists in the artists' communities. Considering that we have two different networks for artists we compared the word cloud of popular artists in the user's communities with both networks. By inspecting the figure 4 we concluded that a significant number of artists belonging to the same community are found among the artists listened to in a community of users. This means users from the same community tend to listen to artists who collaborate or are related (since we found this pattern in both artists' networks). Currently, it is difficult to say if the listeners influence how artists collaborate or if the artists' collaborations impact how listeners gather together. On the other hand, Spotify computes the related artists based on the community's listening history. Given the context, this means that the users' preferences could influence whether the two artists are related. However, one consensus is undeniable; the network of artists reflects the network of users and vice versa.

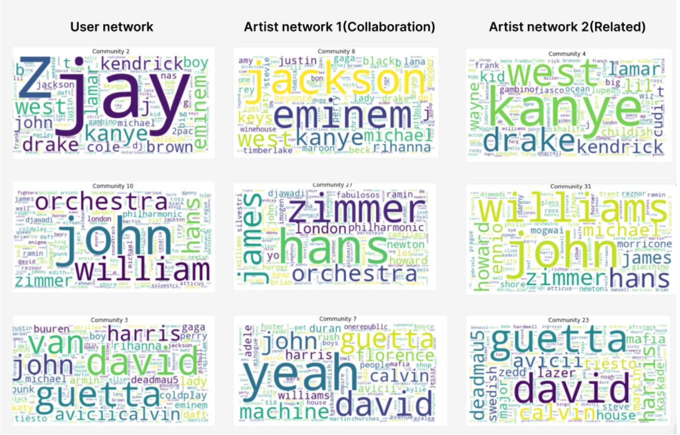


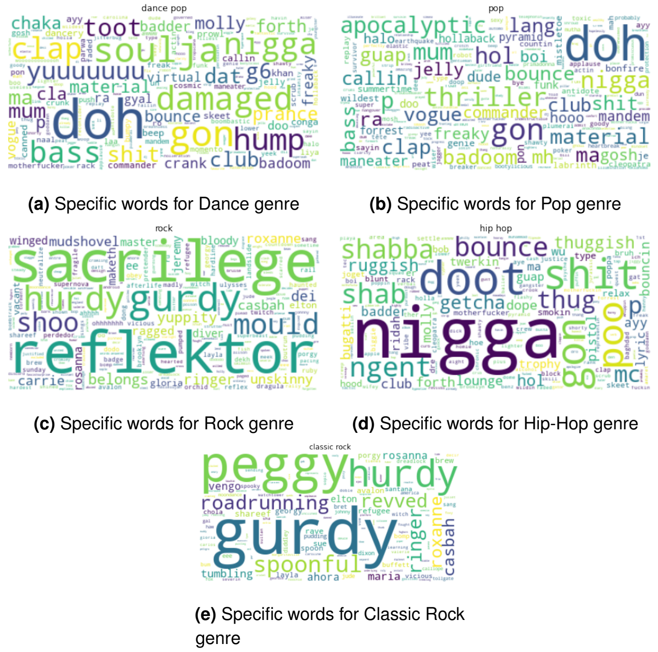
Fig. 4. Word clouds of the popular artists in the communities. The user network is shown in column 1, the collaboration artist network is represented in column 2, and the related artists' network is represented in column 3. Communities, should they share similar artists, can be found in the rows.

The words specific to each genre. The TF-IDF metric was used to find each genre-specific word amongst the most frequent genres in our dataset. We can see that Dance-Pop (Figure 5a) and Pop (Figure 5b) share a significant number of words such as "clap", "club", "bass", which emphasises the similarity between the two. Hip-Hop (Figure 5d) seems to be the best depicted one, where swear words and jargon are ubiquitous, encapsulating its thug-like nature. Rock (Figure 5c) and Classic Rock (Figure 5e) address a larger pallet of themes not necessarily defined by specific words.

The sentiment value by genres. We discovered that the sentiment scores for each of the five most frequent genres appear to be similar. The rarity of words may have contributed to this outcome. As we have seen from the lyrics word clouds of each genre, the displayed words tend to be rare, and they might not be found in the LabMT dataset (which was used for computing the sentiment of each track). However, more negative sentiments can be associated with Hip-Hop which may be attributed to words like "pistol", which have a negative association in the lyrics' word cloud.

Discussion

Over the course of our analysis, we encountered certain limitations. Firstly, we had issues extracting all the unique artists from the given string due to the diverse range of expressions for an artist to indicate the presence of a collaboration with another artist. We used a complex RegEx pattern to detect as



(a) Specific words for Dance genre

(b) Specific words for Pop genre

(c) Specific words for Rock genre

(d) Specific words for Hip-Hop genre

(e) Specific words for Classic Rock genre

Fig. 5. Word clouds which depict the most significant words for each most frequent genres in our dataset

many mainstream ways as possible: 'featuring', 'with', 'and', and so on. However, this has led to a few shortcomings in terms of the artists extracted, which may result in a network not fully representative of the real world. Another issue was that some of the 'artists' extracted were not artists. For example, 'His Orchestra' refers to a specific artist's orchestra, and not a stand-alone artist. This is the trade off from our approach to extracting collaborations from tweets. However, the same confusion was made by Spotify as it considers His Orchestra as an artist but upon closer inspection, all the songs in the playlist are played by an established artist and his orchestra.

Furthermore, given our limited computational resources, it was challenging to sample an appropriate dataset to generate graphs that could generalize to real-world trends and behaviours. Therefore, we applied multiple layers of sampling, such as using a sampling threshold and extracting the largest connected component or the backbone from a graph.

A future extension of our project may include a recommendation system based on community detection, where tracks, artists and genres could be recommended since we discovered that users in a community tend to have similar tastes.

Methods

The explainer notebook can be found at https://github.com/bakos97/Final-project-Social-Graphs/blob/chris/notebooks/Explainer_Notebook.ipynb

Disparity filter. To reduce the density of users' networks, we extracted the backbone using the disparity filter approach, which has been discussed in Extracting the multiscale backbone of complex weighted networks (2). The disparity filter is a way to identify which connections in a network should be preserved. It removes the weakest ones based on a user-defined threshold value. The process is iterative and continues until only the strongest connections remain - the network's backbone.

Louvain's Algorithm. Louvain's algorithm is a popular choice for partitioning communities in a network because it is fast and effective at identifying relatively clear and distinct communities within a network. The algorithm works by iteratively optimizing this measure, starting with each node in its own community and then merging and splitting communities in a way that maximizes modularity. This is useful for a variety of applications, such as identifying groups of users in a social network or understanding how different parts of a complex system are connected.

Given that our network was essentially a social network of artists and users bound together by the music they listened to, we found Louvain's algorithm apt for our use case. We did consider other community detection algorithms like hierarchical clustering algorithm and Girvan-Newman algorithm. However, those algorithms tend to be computationally expensive, especially for large networks with many nodes and edges. We are working with large networks of many nodes, so we decided to prioritize computational efficiency when choosing the algorithm.

Betweenness centrality. Betweenness centrality is a centrality measure that is used in network analysis to identify the importance of a node in a network. This measure is calculated by determining the number of times a node lies on the shortest path between two other nodes in the network. Nodes with a high betweenness centrality are considered to be important because they have the potential to control the flow of information or resources between other nodes in the network.

In the context of our Spotify networks, betweenness centrality could be used to identify the most important users or artists in terms of their ability to connect different listeners or music genres. For example, an artist with a high betweenness centrality in our artist network may be considered a "bridge" between different musical genres or listener demographics, and could be used by the company to recommend music to users or to tailor its algorithms for personalized playlists. Additionally, identifying artists with high betweenness centrality in a Spotify network could be useful for understanding the overall structure and dynamics of the network, and could provide insight into how different musical genres or listeners are connected.

ACKNOWLEDGMENTS. We thank Professor Sune Lehmann Jørgensen and the teaching assistants for their feedback that helped us develop this project.

1. E Zangerle, M Pichl, W Gassler, G Specht, nowplaying music dataset. pp. 21–26 (2014).
2. MÁ Serrano, M Boguñá, A Vespignani, Extracting the multiscale backbone of complex weighted networks. *Proc. Natl. Acad. Sci.* **106**, 6483–6488 (2009).