Comparison of Network Analytics and Significance Analysis on Spotify Artist Feature Collaboration Network

Learning From Networks - Final report

Fabio Cociancich, Luca Fantin, Alessandro Lincetto

Master Degree in Computer Engineering - University of Padova

I. Motivation

This project analyzes the artist collaboration network on Spotify to identify musical trends. The goal is to study artist popularity and their connections using centrality and clustering metrics. The statistical significance of the metrics calculated on the Spotify network is evaluated against random networks, for an in-depth understanding of the dynamics of artistic collaboration. The analysis was performed using the NetworkX and NetworKit packages. The results provide a basis for further research.

II. Dataset

For this project, we used the Spotify Artist Feature Collaboration Network from Kaggle [1]. This dataset consists of a graph where nodes correspond to artists and edges connect artists who have collaborated on at least one song. It has 156,422 nodes, which include around 20,000 artists who appeard in the Spotify weekly charts and around 136,000 artists who had at least one feature with the chart artists, and 300,386 edges between them. Out of the information included with the nodes, the ones we used to analyze our results are the following:

- artist popularity, expressed as an integer number between 0 and 100 (100 corresponding to the most popular artist on the service), according to the Spotify API;
- list of genres, according to the Spotify API.

III. Measures

The measures considered for these analyses are both graph- and node-level graph metrics. At the graph level, we compute the number of connected components and the clustering coefficient of the graph, both global and average. At the node level, instead, we have the local clustering coefficients and a series of centrality measures. Alongside those presented during the lectures (degree, closeness, betweenness, PageRank), we also considered the *eigenvector centrality*, which is built on the intuition that a node is important if it is connected to other important nodes. Given a graph $G = (V, E)$, let us define $\mathbf{x} \in \mathbb{R}^{|V|}$ the vector of the centrality values for all nodes in G, A the adjacency matrix of G and $\lambda \neq 0$ a constant. For any node i we can write:

$$x_i = \frac{1}{\lambda} \sum_{j=1}^{|V|} A_{i,j} x_j \quad \rightarrow \quad Ax = \lambda x$$

The mathematical representation of the intuition can thus be reformulated as finding the eigenvector of the adjacency matrix corresponding to the eigenvalue λ; such vector includes the values of the eigenvector centrality for all nodes. This centrality measure has been studied extensively [2] [3] [4], also in the context of social media network analysis [5], including Spotify [6] [7].

IV. Significance analysis framework

The significance analysis we performed is composed by the statistical testing framework and the random graph model chosen. For the latter, we chose the Holme-Kim model [8]. The generation of a random graph starts from a number of nodes smaller than the desired graph size and no edges, iteratively adds new nodes and connects them with already existing ones with a distribution that favours nodes with

an already high degree. The resulting graphs show a power-law distribution of the degrees: the probability of seeing a node with a certain degree decreases exponentially as the degree increases. Such characteristic is observed in many real-world networks and is captured by this model more accurately compared to the Erdős-Rényi model. Furthermore, this model produces graphs with tunable global clustering coefficients by creating additional edges: once a newly created node v is connected to an existing one w, a new edge is created between v and one of the neighbours of w with a certain probability.

The null hypothesis considered in this project states that the metric values computed on the real graph well conform to the distribution determined by the generated random graphs, which implies our Spotify dataset does not have any significant feature that can explain the values we compute. Our statistical testing procedures consists of two steps. Any statistical test assumes that the considered population has a known distribution, most often a Gaussian one. Because of the specificity of our random graph model, the first step computes how similar the distribution of the metric we are considering is compared to a Gaussian distribution, through a *normality test*. We used Shapiro-Wilk test [9], since it is considered the most powerful normality test available [10]. The second step then checks the validity of our null hypothesis by computing the probability that the random distribution of the metric generates a value greater or lower than the real value, also called p-value.

V. Code

All the code developed for this project, together with results files, dataset files and more, can be found in our GitHub repository [11]. The central script is `main.py`, which computes any combination of the measures presented in section III on various graphs, depending on the command line arguments provided. The script can work on the entire dataset, subgraphs taken from the dataset with respect to certain genres or popularity thresholds, and random graphs generated with the Holme-Kim model [8] with parameters specified through the command line arguments. The libraries NetworkX and NetworKit are used to represent the graphs [12], compute the metrics [13] [14] [15] and generate the random graphs [16].

VI. Experimental setup

All computations on graphs have been performed on the CAPRI High-Performance Computing (HPC) cluster [17]. The hardware capabilities and the presence of the SLURM job scheduler system allowed us to perform heavy computations in a feasible time frame. This system features the following hardware:

- 16 Intel(R) Xeon(R) Gold 6130 @ 2.10GHz CPUs
- 6 TB DDR4 RAM
- 2 NVIDIA Tesla P100 16GB GPUs
- 40 TB of disk space

On the other side, the analysis of the data computed by the cluster has been performed on our local machines. They all employ AMD Ryzen 5/7 CPUs and RAMs ranging from 8 GBs and 24 GBs.

VII. Data analysis on whole dataset

In the first part of the analysis, artist rankings were calculated and sorted based on two key centrality metrics: degree centrality and closeness centrality. Degree centrality identified artists with the highest number of direct connections (i.e., those who collaborate with the most artists), so these artists occupy the top positions in the ranking. Some of the top-ranked artists include Johann Sebastian Bach, Traditional, Mc Gw, MC MN, and Jean Sibelius, who stand out for their high number of collaborations. Closeness centrality, on the other hand, highlighted artists who are more "central" in the network, meaning those who, despite having fewer direct collaborations, are well-positioned to interact with the entire network. Artists such as R3HAB, Snoop Dogg, Diplo, David Guetta, and Tiësto are at the top of this ranking, indicating their global influence within the network.

We have created a visual representation of the distribution of centrality measures for each artist. The resulting graph clearly shows how some measures, such as degree centrality, are

widely distributed, with many nodes having a low number of collaborations and a few nodes having a very high number. Closeness centrality, on the other hand, tends to concentrate within a narrow range, suggesting that many artists are relatively close in the network, while some are significantly distant from the rest of the graph. Betweenness, PageRank, and eigenvector centrality show a similar distribution, with many nodes having low values and a few emerging with higher values, indicating that there are artists with very high influence in the network, but they are few. The comparison among the different centrality measures highlights how each of them measures different aspects of the structure and dynamics of the network.

Following this, the average ranking for each artist was calculated, based on the combination of rankings obtained through the various centrality metrics (degree, closeness, betweenness, PageRank, eigenvector centrality). This average ranking provided an overall view of the importance of each artist within the network, taking into account all dimensions of centrality. Artists like Snoop Dogg, Gucci Mane, and David Guetta emerged as the most important based on this average ranking, consistently ranking high across various metrics.

A complete ranking of the artists was finally calculated, based on all the centrality measures analyzed: degree, closeness, betweenness, PageRank, eigenvector centrality, and an average ranking that integrates the results of all the metrics. Betweenness centrality identifies artists who act as "bridges" between different areas of the network, such as Snoop Dogg, Gucci Mane, and David Guetta, while PageRank and eigenvector centrality highlight artists connected to important nodes, boosting their centrality value. The average ranking provides a comprehensive summary of the artists' importance, combining all the metrics. The analysis shows that some artists, such as Snoop Dogg and David Guetta, rank high in multiple rankings, suggesting their central and influential role in the collaboration network.

VIII. Data analysis on subgraphs

IX. Data analysis on random graphs

For the significance analysis we considered the real graph and a selection of subgraphs. Because of the NetworkX implementation of the Holme-Kim, these graphs had to have more edges than nodes and a global clustering coefficient lower than 0.3 to be accurately resembled by the random graphs. For each of these graphs, 200 random graphs were generated, with the same number of nodes, a similar number of edges and a comparable global clustering coefficient. The last part was easy for graphs with a coefficient under 0.1, but reaching values above that threshold became almost impossible. On these generated graphs, we computed all available graph-level metrics and the maximum and average value of the closeness and eigenvector centralities, in order to represent both distributions of the node rankings with respect to centrality values. The closeness centrality was not computed for the larger graphs, corresponding to the entire dataset and the pop genre subgraph, due to time constraints.

A. Normality test

For each set of random graphs generated, all values of the metrics were tested against a Gaussian distribution using the Shapiro-Wilk test. The result, reported in table I, was that most of the distributions could be assimilated to a Gaussian one. For the other ones, plotting the histogram of the values revealed a distribution graphically resembling a Gaussian curve. Thus, we decided to take into consideration all metrics for all sets of graphs for the next phase of statistical testing.

B. p-value computation

For each metric listed in this section, we took its value from the real (sub)graph and computed the probabilities that the Gaussian distribution corresponding to the random graphs give a value higher or lower than the real one. The results, reported in table II, show that we cannot accept our null hypothesis, thus we can say that our (sub)graphs have particular features that affect the metric values in some significant

way. In fact, for almost all graphs, if the real ones were generated by the same distribution as the random graphs, we should expect to see higher maximum and average values for the considered centralities, a higher average clustering coefficient and a lower global clustering coefficient. The latter results confirms the difficulties in replicating the value computed on the real graphs through the random graph model. The results for the other metrics could be due to the more fragmented nature of the real graph, which has a large number of connected components and less edges.

X. Conclusion

A. Graph analysis

B. Genre subgraphs analysis

We analyzed also some subgraphs created considering only a particular genre. The genres with highest clustering coefficients are "latin" (0.165 avg. cc, 0.300 global cc.) and "trap" (0.189 avg. cc, 0.270 global cc.).

The ones with lowest clustering coefficients are "techno" (0.0017 avg. cc, 0.0029 global cc.) and "classical" (0.0013 avg. cc, 8.7 e-05 global cc., 1260 nodes, 775 edges, 541 connected components).

C. Popularity subgraphs analysis

By analysing the subgraphs created considering only the 0.1% most popular artists we can note that it has quite high clustering coefficients (0.277 avg. cc, 0.363 global cc.).

Contributions

References

[1] Julian Freyberg. *Spotify Artist Feature Collaboration Network*. URL: https://www.kaggle.com/datasets/jfreyberg/spotify-artist-feature-collaboration-network.

[2] Phillip Bonacich. "Some unique properties of eigenvector centrality". In: *Social Networks* 29.4 (2007), pp. 555–564. ISSN: 0378-8733. DOI: https://doi.org/10.1016/j.socnet.2007.04.002.

[3] Stephen Borgatti, Kathleen Carley, and David Krackhardt. "On the Robustness of Centrality Measures Under Conditions of Imperfect Data". In: *Social Networks* 28 (May 2006), pp. 124–136. DOI: 10.1016/j.socnet.2005.05.001.

[4] Leo Spizzirri. "Justification and application of eigenvector centrality". In: *Algebra in Geography: Eigenvectors of Network* (2011). URL: https://sites.math.washington.edu/~morrow/336_11/papers/leo.pdf.

[5] Warih Maharani, Adiwijaya, and Alfian Akbar Gozali. "Degree centrality and eigenvector centrality in twitter". In: *2014 8th International Conference on Telecommunication Systems Services and Applications (TSSA)*. 2014, pp. 1–5. DOI: 10.1109/TSSA.2014.7065911.

[6] Tobin South, Matthew Roughan, and Lewis Mitchell. "Popularity and centrality in Spotify networks: critical transitions in eigenvector centrality". In: *Journal of Complex Networks* 8.6 (Mar. 2021), cnaa050. ISSN: 2051-1329. DOI: 10.1093/comnet/cnaa050. URL: https://doi.org/10.1093/comnet/cnaa050.

[7] Tobin South. "Network analysis of the Spotify artist collaboration graph". In: *Australian Mathematical Sciences Institute* (2018), pp. 1–12. URL: https://vrs.amsi.org.au/wp-content/uploads/sites/84/2018/04/tobin_south_vrs-report.pdf.

[8] Petter Holme and Beom Jun Kim. "Growing scale-free networks with tunable clustering". In: *Physical Review E* 65.2 (Jan. 2002). ISSN: 1095-3787. DOI: 10.1103/physreve.65.026107. URL: http://dx.doi.org/10.1103/PhysRevE.65.026107.

[9] S. S. Shapiro and M. B. Wilk. "An Analysis of Variance Test for Normality (Complete Samples)". In: *Biometrika* 52.3/4 (1965), pp. 591–611. ISSN: 00063444, 14643510. URL: http://www.jstor.org/stable/2333709 (visited on 12/17/2024).

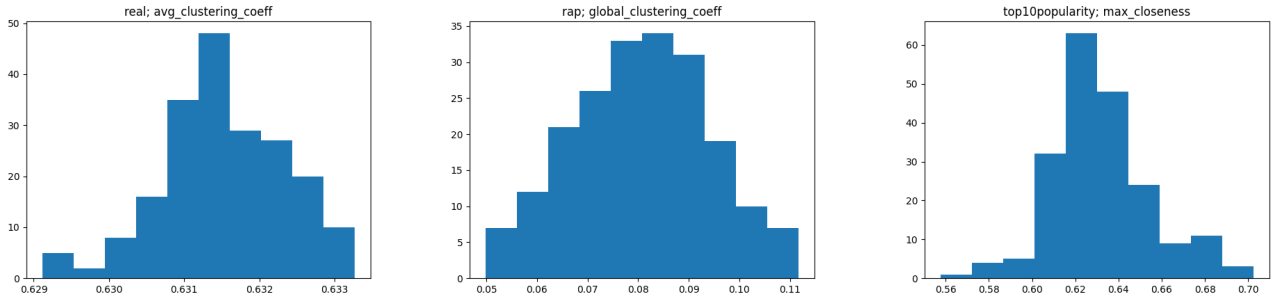
[10] Nornadiah Mohd Razali and Bee Yap. "Power Comparisons of Shapiro-Wilk, Kolmogorov-Smirnov, Lilliefors and

Reference graph	*Average cc*	*Global cc*	*Approximate global cc*	*Maximum eigenvector*	*Average eigenvector*	*Maximum closeness*	*Average closeness*
House subgraph	✗	✓	✓	✓	✓	✓	✓
Pop subgraph	✗	✗	✗	✓	✓		
Rap subgraph	✗	✗	✗	✓	✓	✓	✗
Whole dataset	✗	✗	✗	✓	✓		
Top 10% popularity subgraph	✗	✗	✗	✓	✓	✗	✗
Trap subgraph	✗	✗	✗	✓	✓	✓	✗

TABLE I: Results of the Shapiro-Wilk normality tests for all considered graphs. The "reference graph" is the graph to which the random graphs used in the analysis refer to. "cc" stands for "clustering coefficient".

Reference graph	*Average cc*	*Global cc*	*Approximate global cc*	*Maximum eigenvector*	*Average eigenvector*	*Maximum closeness*	*Average closeness*
House subgraph	>	<	<	>	>	>	>
Pop subgraph	>	<	<	>	>		
Rap subgraph	>	<	<	>	>	>	>
Whole dataset	>	<	<	>	<		
Top 10% popularity subgraph	>	<	<	>	>	>	>
Trap subgraph	>	<	<	>	>	>	>

TABLE II: Results of the p-value computations: the contents of the cells represent how we should expect the metric values to be, compared to the values computed on the real (sub)graphs, if we were to accept our null hypothesis. "cc" stands for "clustering coefficient".



(a) Whole dataset, average clustering coefficient

(b) Rap subgraph, global clustering coefficient

(c) Top 10% popularity subgraph, maximum closeness

Fig. 1: Example of histograms for the metrics computed on random graphs that do not have a Gaussian distribution according to the Shapiro-Wilk test.

Anderson-Darling Tests". In: *J. Stat. Model. Analytics* 2 (Jan. 2011).

[11] Fabio Cociancich, Luca Fantin, and Alessandro Lincetto. *lfn_project*. URL: https://github.com/fantinluca/lfn_project/.

[12] *NetworkX, graph generators*. URL: https://networkx.org/documentation/stable/reference/generators.html.

[13] *NetworkX, centrality algorithms*. URL: https://networkx.org/documentation/stable/reference/algorithms/centrality.html.

[14] *NetworKit, local clustering coefficient*. URL: https://networkit.github.io/dev-docs/python_api/centrality.html#networkit.centrality.LocalClusteringCoefficient.

[15] *NetworKit, global clustering coefficient*. URL: https://networkit.github.io/dev-docs/python_api/globals.html#networkit.globals.ClusteringCoefficient.

[16] *NetworkX, Holme-Kim model*. URL: https://networkx.org/documentation/stable/reference/generated/networkx.generators.random_graphs.powerlaw_cluster_graph.html.

[17] University of Padova Strategic Research Infrastructure Grant 2017. *CAPRI: Calcolo ad Alte Prestazioni per la Ricerca e l'Innovazione*. URL: https://capri.dei.unipd.it/.