Community Detection of Singers in Spotify Rock Playlists Using Louvain Method

Annisa Cahyani Surya[a1], Christyan Tamaro Nadeak[a2]
[a]Data Science Study Program, Faculty of Science, Institut Teknologi Sumatera, Indonesia
e-mail: [1]annisa.121450114@student.itera.ac.id, [2]christyan.nadeak@sd.itera.ac.id

Abstrak

Spotify merupakan salah satu platform streaming musik yang memungkinkan pengguna membuat playlist dan memilih lagu sesuai preferensi mereka. Aliran musik rock telah lama hadir dan menjadi salah satu genre pilihan utama di Spotify. Di Indonesia, musik rock memiliki sejarah panjang, di mana aliran ini tidak hanya berfungsi sebagai hiburan, tetapi juga sebagai medium perjuangan sosial dan politik. Penelitian ini bertujuan untuk memahami bagaimana preferensi pengguna membentuk pola keterhubungan antar artis. Dengan menerapkan metode deteksi komunitas Louvain, artis dikelompokkan berdasarkan keterhubungannya dalam playlist, sehingga dapat diungkap pola komunitas pada genre rock. Data penelitian diperoleh melalui proses scraping API Spotify dengan mengumpulkan playlist menggunakan kata kunci "Rock Indonesia". Hasil penelitian menunjukkan bahwa konfigurasi optimal diperoleh pada k-core 3 dengan threshold bobot 0.39, menghasilkan nilai modularitas sebesar 0.4782. Tiga komunitas utama berhasil diidentifikasi, masing-masing dikelompokkan berdasarkan subgenre, periode aktif, dan label rekaman.

Kata kunci: *Deteksi Komunitas, Algoritma Louvain, Rock, Playlist*

Abstract

Spotify is one of the leading music streaming platforms, allowing users to create playlists and select songs based on their preferences. Rock music has remained a prominent genre on Spotify, especially in Indonesia, where it holds historical and cultural significance and serves as a medium for social and political expression. This study investigates how user preferences shape the network structure among Indonesian rock artists. Using the Louvain community detection method, artists were grouped based on their co-occurrence in playlists to uncover community patterns within the genre. Data were collected by scraping playlists using the keyword "Rock Indonesia." The optimal configuration was found with a k-core value of 3 and an edge weight threshold of 0.39, resulting in a modularity score of 0.4782. Three main communities were identified, differentiated by subgenre, active period, and record label.

Keywords : *Community Detection, Louvain Algorithm, Rock, Playlist*

1. Introduction

The development of digital music streaming platforms has revolutionized the way people access and consume music. Spotify is one of the leading music streaming platforms that provides a playlist feature, enabling users to organize their favorite songs based on individual preferences. The presence of this feature not only encourages personalized music consumption but also reveals general listening trends toward particular artists or music genres. One of the music genres that frequently ranks among the top choices on Spotify is rock music. To date, rock music remains enduring and maintains a strong fan base, as evidenced by its consistent presence among the top ten most frequently appearing genres in Spotify playlists.

In Indonesia, rock music holds significant historical and cultural value. Since its introduction in the 1950s, the genre has undergone various social dynamics, including being banned during President Soekarno's administration due to its perceived contradiction with national culture. However, rock music experienced rapid growth in the late 1990s, when it became a medium for youth expression in response to social and political changes, particularly during the transition toward democracy. According to Jeremy Wallach, an anthropologist specializing in Southeast Asian popular music, rock music in the late 1990s functioned not only as entertainment but also as a medium for identity formation and social expression [1]. Given this background, the rock genre represents a relevant case for deeper investigation, particularly in understanding how user preferences on digital platforms shape relationships among artists.

Despite the large and consistent fan base of rock music, studies examining interaction patterns among artists based on user preferences remain limited. Teirelbaum et al. [2] analyzed interactions among artists in social networks and their implications for musical similarity and collaboration, but did not specifically focus on the rock genre. Meanwhile, Taufik [3] applied a Social Network Analysis (SNA) approach to examine relationships among users in digital music platforms based on preference similarity, but did not explore artist-to-artist relationships formed from such preferences.

SNA offers various methods for detecting communities within social networks. One of the most efficient and widely used methods is the Louvain method, which is known for its ability to optimize modularity values and its stability in large-scale networks [4]. Motschnig et al. [5] reported that this method demonstrates consistent performance across various network sizes. Several studies have applied this method in the music domain, such as identifying communities among metal music subgenres and understanding fan perceptions [6], as well as clustering users based on similar musical preferences [3].

However, to date, no study has explicitly mapped rock artist communities on Spotify using a Louvain-based approach derived from user playlists. Therefore, this study aims to detect communities of rock artists based on their co-occurrence within Spotify user playlists. Through this approach, the study seeks to identify patterns of community formation within the rock genre based on playlists created by users.

2. Research Method / Proposed Method

2.1. Dataset

The data used in this study were obtained through a scraping process from Spotify. Data collection was conducted on public playlists containing the keywords “Rock OR Indonesia.” A total of 500 playlists were randomly selected as the sample. The collected information includes the username, playlist title, artist name, and song title.

2.2. Data Preprocessing

At this stage, several filtering procedures were applied to ensure that only relevant and high-quality playlists were included in the analysis. The preprocessing steps consisted of the following:

1. Only playlists containing the keyword “Rock” in the title were retained to ensure relevance to the research topic.
2. Playlists containing fewer than five songs were removed, as they were considered insufficient to represent meaningful music preferences. Additionally, playlists containing songs from only a single artist were excluded, since they do not allow the formation of inter-artist relationships within the network.
3. Filtering was performed based on artist popularity, where only artists appearing in at least 10 playlists were retained. This step aimed to reduce noise from infrequently occurring artists in the dataset.
4. Playlists not originating from Indonesia were removed based on their titles.
5. Automatically generated playlists created by Spotify were excluded to ensure that the network reflects individual user preferences only.

In addition to the filtering procedures described above, duplicate records were removed during preprocessing to ensure greater graph accuracy. In the constructed network, each node represents a unique artist, and each edge represents the co-occurrence of distinct artists within a playlist without duplication.

2.3. Bipartite Graph Construction Between Users and Artists

In representing relationships between two distinct types of entities, bipartite graphs are commonly employed. A bipartite graph is defined as a graph whose vertex set can be partitioned into two disjoint subsets, denoted as X and Y, such that every edge connects a vertex in X to a vertex in Y.

In the context of this study, each user is assumed to have one playlist; therefore, each playlist represents a unique user, and vice versa. The dataset, originally structured in tabular form, is mapped into a bipartite graph where user nodes constitute set X and artist nodes constitute set Y. An edge is established between a user node and an artist node if the user includes the artist in their playlist.

Each row in the dataset is transformed into an edge in the bipartite graph. The resulting network structure is represented using either an adjacency list or an adjacency matrix. The adjacency matrix encodes the connectivity between nodes in binary form, where a value of 1 indicates the presence of an edge and 0 indicates the absence of a connection, as shown in Figure 1.

	user1	user2	artis1	artis2	artis3
user1	0	0	1	1	0
user2	0	0	1	1	1
artis1	1	1	0	0	0
artis2	1	1	0	0	0
artis3	0	1	0	0	0

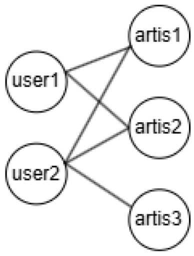

Figure 1. Adjacency matrix node user and artist, and the bipartite graph

2.4. Projection to a Unipartite Graph

In several applications, bipartite graphs are converted into unipartite networks consisting of nodes from only one group, where the relationships between nodes are based on their connections through nodes from the other group [7].

At this stage, the bipartite graph is transformed into a unipartite graph to focus on artist-to-artist relationships within rock playlists. The projection is performed by mapping set Y(artist nodes), where two artist nodes are connected if they share common neighbors in set X(user nodes). The edge weight between two artist nodes is calculated based on the number of shared neighbors in set X. Therefore, in the unipartite graph, the nodes represent artists, while the edges indicate the number of users who added both artists to the same playlist, as shown in Figure 2.

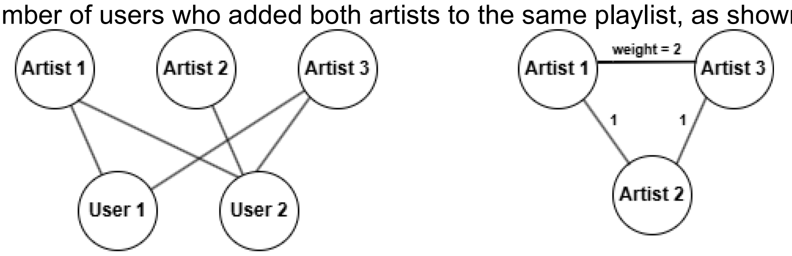

Figure 2. Bipartite graph converted into Unipartite graph

2.5. Weight Transformation Using Jaccard Similarity

To represent the strength of relationships between artists more proportionally, a weight transformation was performed using the Jaccard Similarity approach. This method measures the level of similarity between two nodes based on the number of shared neighbors relative to the total number of their unique neighbors. In the context of the unipartite artist network, Jaccard similarity is used to evaluate how frequently two artists appear in the same playlists relative to the total number of playlists containing at least one of them.

The Jaccard Similarity between two artists u and v is defined as follows [8]:

$$\mathrm{J(u,v)}=\frac{|\mathrm{N(u)} \cap \mathrm{N(v)}|}{|\mathrm{N(u)} \cup \mathrm{N(v)}|}$$

where $N(u)$ and $N(v)$ denote the sets of playlists in which each artist appears. Although the initial weight—calculated based on the frequency of co-occurrence within playlists—provides basic information regarding inter-artist relationships, this approach does not fully capture contextual similarity between artists in the network. By applying Jaccard similarity, the resulting weights are normalized within the range [0,1], thereby providing a more balanced and meaningful measure of association strength between artists based on their co-occurrence in user playlists.

2.6. Louvain Algorithm

The Louvain algorithm is a greedy heuristic-based method designed to partition a graph into communities by maximizing modularity. It is widely recognized for its efficiency and effectiveness in detecting communities in large-scale networks [4].

The algorithm operates iteratively through two main phases: (1) local modularity optimization, where each node is reassigned to neighboring communities to maximize modularity gain, and (2) community aggregation, where identified communities are treated as new nodes and the optimization process is repeated until no significant improvement in modularity is achieved.

After constructing the unipartite artist network and transforming edge weights using Jaccard Similarity, community detection was performed using the Louvain algorithm. Prior to its application, two filtering steps were conducted to enhance network structure: (1) edge-weighted thresholding to remove weak connections and reduce noise, and (2) k-core decomposition to retain only nodes with a minimum degree, focusing the analysis on the structurally stable core of the network [9]. The Louvain algorithm was then applied iteratively until an optimal modularity value was obtained.

2.7. Community Evaluation

To assess the extent to which the community partition generated by the Louvain algorithm represents a meaningful and well-defined network structure, evaluation was conducted using the modularity metric. Modularity is a widely used measure for quantifying the strength of a network's division into communities. The theoretical range of modularity values lies between −1 and 1, with typical values ranging from 0.3 to 0.7 [10]. In this study, a modularity threshold of 0.5 was adopted, referring to the recommendation by Newman and Girvan [11], which suggests that such a value indicates a structurally strong community structure.

In weighted networks, modularity (Q) is mathematically defined as follows [11], [12]:

$$\mathrm{Q} = \frac{1}{2m}\sum_{i,j}\left[A_{ij} - \frac{k_i k_j}{2m}\right]\delta_{(ci,cj)}$$

3. Result and Discussion

3.1. Data Exploration

From the data collection process conducted through scraping using the Spotify API, a total of 427 public Spotify playlists were obtained, comprising 29,309 artist–song records. An analysis was then performed to determine the average number of songs per playlist. Among the 427 playlists collected, the average number of songs per playlist was 62. The minimum number of songs in a playlist was 1, while the maximum reached 434 songs. The distribution of the number of songs per playlist is presented in Figure 3.

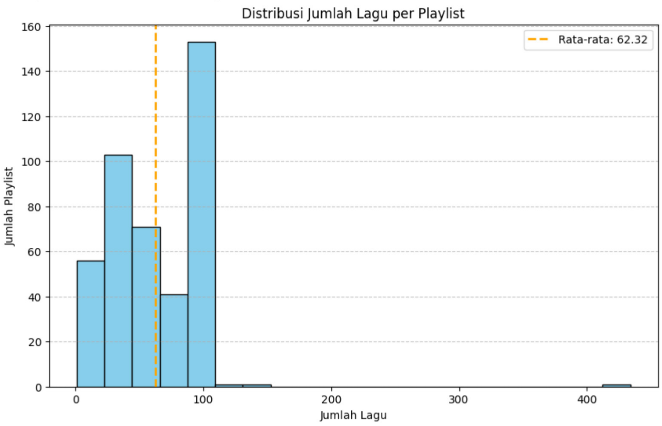

Figure 3. Song Distribution

3.2. Data Preprocessing

The data preprocessing stage was conducted to ensure that only relevant and clean playlists were used in the network analysis. The filtering steps and their results are described as follows:

1. **Filtering Playlists Based on the Keyword “Rock”**: From the total of 427 public Spotify playlists collected, 269 playlists met this criterion, namely those containing the keyword “rock” in their titles.
2. **Removing Playlists with Fewer than Ten Songs and Playlists Containing Only One Artist**: Playlists containing fewer than five songs were removed, as they were considered insufficient to represent strong music preferences. After this filtering stage, 255 playlists satisfied the criteria, namely those containing more than ten songs and featuring more than one artist.
3. **Retaining Artists Appearing in at Least 10 Playlists**: This filtering step was performed to reduce noise from artist nodes that rarely appeared in the dataset. After this process, 211 playlists remained.
4. **Removing Playlists Not Originating from Indonesia**: Playlists containing titles indicating origins or music styles not associated with Indonesia were removed to retain only Indonesian rock artists. This filtering resulted in 137 playlists.
5. **Removing Spotify-Generated Playlists**: Playlists created automatically by Spotify do not reflect individual user preferences; therefore, such playlists were removed. The number of playlists remained at 137 after this step.
6. **Removing Artists from Outside Indonesia**: This process involved verifying each unique artist’s country of origin. From the 137 selected playlists, 186 unique artists were identified. After verification, 72 artists were confirmed to originate from Indonesia.

Duplicate removal during preprocessing was performed to eliminate duplicate users. In this study, each user was represented by only one playlist to ensure unique preference representation. From the 122 playlists that passed the previous filtering stage, it was found that two users had created more than one playlist. After removing these duplicates, 120 playlists from 120 unique users remained. The final dataset consists of 72 artists and was used for graph construction in the subsequent stage.

3.3. Bipartite Graph Construction Between Users and Artists

Using the NetworkX library in Python, the dataset was transformed into a bipartite graph, where edges represent the relationships between users and the artists included in their playlists. The resulting bipartite graph consists of 192 nodes and 1,324 edges. These nodes comprise 120 unique users (represented by light blue nodes) and 72 artists (represented by pink nodes), all of which had previously undergone the filtering process. The visualization of the bipartite graph is shown in the Figure 4.

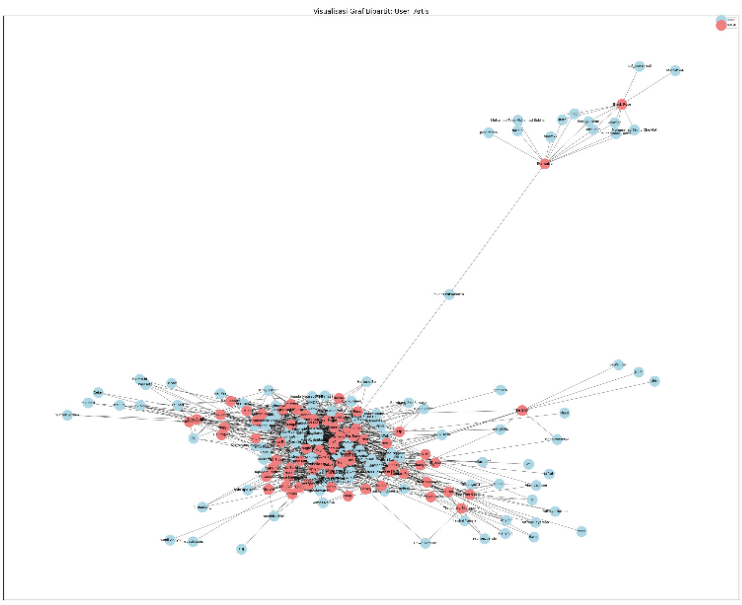

Figure 4. Bipartite Graph Users and Artists

3.4. Projection to a Unipartite Graph

To enable the application of the Louvain community detection algorithm, the network structure was transformed into a unipartite graph consisting of a single type of node. Therefore, the previously constructed bipartite graph was projected into a weighted unipartite graph representing artist-to-artist relationships. In this projection process, only nodes representing artists were retained. Two artists were connected by an edge if they appeared in the same playlist. The weight of each edge reflects the frequency of co-occurrence, namely the number of playlists containing both artists simultaneously. Thus, the edge weight represents the level of association between artists in the context of user preferences.

The projection resulted in a weighted unipartite graph consisting of 72 nodes (artists) and 2,187 edges. This graph serves as the basis for community detection using the Louvain algorithm to identify community structures within the Spotify rock music ecosystem. The visualization of the projected graph is shown in Figure 5.

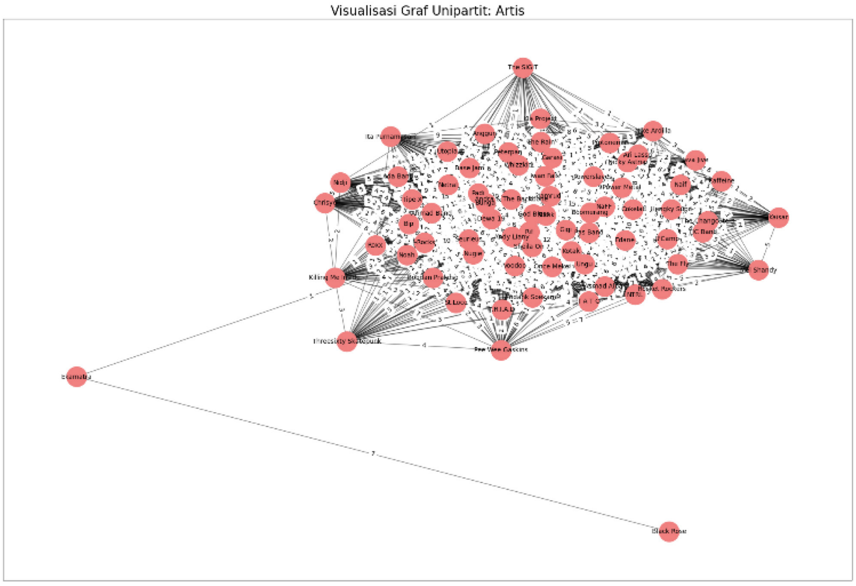

Figure 5. Unipartite Graph Artists

3.5. Results and Evaluation

To evaluate the effect of edge filtering on the quality of the community structure within the Spotify rock artist network, a series of experiments was conducted by applying various threshold values to the weights computed using Jaccard similarity. The tested threshold values ranged from θ = 0.15, 0.20, 0.25 up to 0.40. The primary objective of applying these thresholds was to eliminate weak connections that could introduce noise into the network, thereby improving the modularity value and producing a more well-defined community structure.

In addition to edge thresholding, node filtering was performed using the k-core decomposition technique with values of k = 3, 4, 5. The combination of threshold and k values was used to observe their influence on community detection performance, particularly in terms of the resulting modularity values. The experimental results are presented in Figure 6.

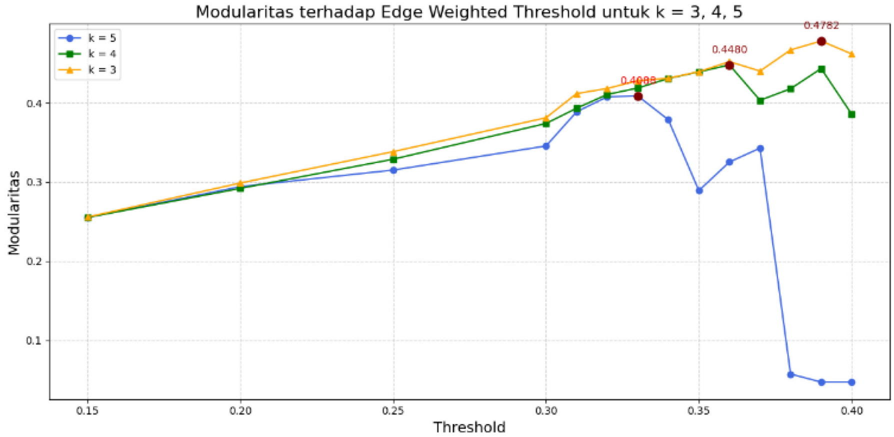

Figure 6. The relationship between the edge-weight threshold and modularity for different k-core values (k=3,4,5). The red points indicate the maximum modularity obtained for each value of k

In general, increasing the threshold from 0.15 to approximately 0.33 leads to an increase in modularity across all three configurations. This occurs because weak or random connections are filtered out, resulting in a clearer and more well-defined community structure. However, when the threshold is increased excessively, modularity decreases sharply. This decline occurs because excessive edge removal makes the network too sparse, thereby hindering optimal community detection. This finding is consistent with Fortunato [13], who stated that overly sparse networks lose the connection density necessary for community formation.

In addition to the threshold, the k value in k-core decomposition also influences modularity. For k = 3, the highest modularity value of 0.4782 was achieved at θ = 0.39, as the number of retained nodes and connections remained sufficient to form solid communities. In contrast, for k = 5, only highly connected nodes were retained, making the graph more sensitive to edge removal. As a result, the maximum modularity was lower, at 0.4088 achieved at θ = 0.33, and decreased more rapidly thereafter.

The highest modularity was obtained under the k-core 3 configuration, with a modularity value of 0.4782 at θ = 0.39. Under this highest-modularity network structure, three main communities were identified, distinguished by different colors, as shown in Figure 7.

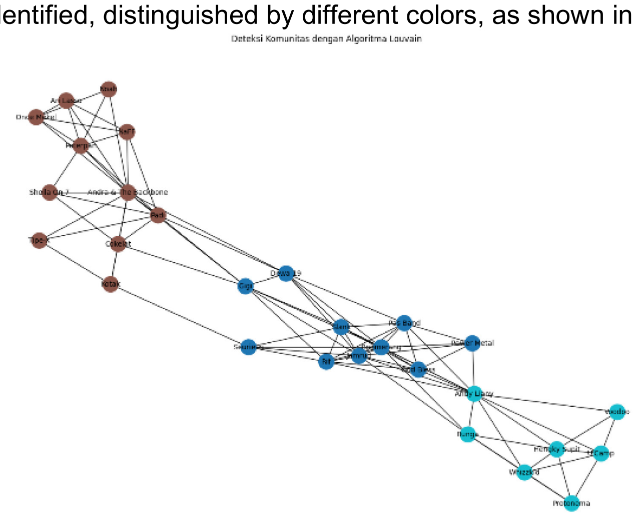

Figure 7. Graph with modularity 0.4782

Although the modularity value obtained from this configuration does not exceed the 0.5 threshold—commonly considered an indicator of a clear and significant community structure—several previous studies have shown that modularity values above 0.3 may already indicate meaningful community structures in complex networks. In studies [14], Newman found that in many real-world network applications, modularity values typically range from 0.3 to 0.7. The modularity value of 0.4782 obtained in this study falls within this range and can therefore be considered to reflect a reasonably strong community division.

To strengthen this interpretation, detailed information regarding the members of each community is presented in Table 1. An analysis of community membership reveals a tendency toward grouping based on active period, musical subgenre, and record label. Based on the table, Community 1 (colored red) consists of 11 nodes, predominantly bands from the late 1990s to early 2000s associated with pop rock and alternative rock subgenres. Community 2 (colored dark blue) includes 10 nodes, mostly bands from the 1980s to early 1990s characterized by hard rock and heavy metal subgenres. Meanwhile, Community 0 (colored light blue) comprises 7 nodes, consisting of singers and bands from the late 1980s to mid-1990s, representing more diverse rock subgenres with some overlapping record labels.

Thus, although the modularity value obtained in this study does not reach the ideal threshold of 0.5, the findings from previous studies and the analysis of community characteristics suggest that the resulting network structure reflects meaningful connectivity patterns, both in terms of the modularity value and the identifiable characteristics of artists within each community.

Table 1. Results of Community

Komunitas	***Jumlah Node***	***Artis***
Komunitas 0	7	Protonema, Whizzkid, Bunga, Voodoo, U'Camp, Hengky Supit, dan Andy Liany
Komunitas 1	11	Noah, Sheila On 7, Ari Lasso, Tipe-X, Dewa 19, Gigi, Once Mekel, Peterpan, Cokelat, NaFF, Kotak, Padi, dan Andra & The Backbone
Komunitas 2	10	Slank, Rif (Rythim in Freedom), Jamrud, Seurieus, Pas Band, Boomerang, God Bless, dan Power Metal

4. Conclusion

Based on the testing and analysis conducted in this study, it can be concluded that the application of the Louvain algorithm successfully partitioned the rock artist network into three main communities, achieving the highest modularity value (0.4783) under the configuration k = 3 and θ = 0.39. This result indicates a significant community structure with optimal modularity under the selected parameters.

Furthermore, the resulting network structure reflects meaningful connectivity patterns through the characteristics of artists within each community. The communities are differentiated based on artists' active periods, musical subgenres, and record label affiliations. Community 1 is predominantly composed of pop rock and alternative rock bands from the late 1990s to early 2000s. Community 2 consists mainly of hard rock and heavy metal bands from the 1980s to 1990s era. Meanwhile, Community 0 includes singers and bands from the late 1980s to mid-1990s, representing more diverse rock subgenres with some overlapping record labels. These findings demonstrate that the Louvain algorithm is capable of uncovering natural connectivity patterns among artists within Spotify user playlists.Contains statements to answer the problems described in the introduction and suggestions for further research if needed in accordance with the results of the study.

References

[1] Wallach J., *Rock in Indonesia*. In: Kahn JS. Editor. Southeast Asian Identities: Culture and the Politics of Representation in Indonesia, Malaysia, Singapore, and Thailand. Singapore: ISEAS Publishing; 2008: 109–137.

[2] Teitelbaum T, Balenzuela P, Cano P, Buldu J. *Community structures and role detection in music networks*. Chaos. 2009; 18: 043105.

[3] Taufik R, Muhaqiqin, Sholehurrohman R, Ilman IS. *Analisis jaringan pengguna platform musik dalam membentuk tren musik*. Publikasi Elektronik Pengembangan Aplikasi Digital Untuk Negeri. 2022; 3(2): 296–305.

[4] Blondel V, Guillaume JL, Lambiotte R, Lefebvre E. *Fast unfolding of communities in large networks*. Journal of Statistical Mechanics: Theory and Experiment. 2008; 2008: P10008.
[5] Motschnig N, Ramharter A, Schweiger O, Zabka P, Foerster KT. *On comparing and enhancing common approaches to network community detection*. CoRR. 2021.
[6] Jiang Z, Huynh H. *Unveiling music genre structure through common interest communities*. Social Network Analysis and Mining. 2022; 12: 35.
[7] Banerjee S, Jenamani M, Pratihar D. *Algorithms for projecting a bipartite network*. 2017 10th International Conference on Contemporary Computing (IC3). Noida. 2017: 1–3.
[8] Tan PN, Steinbach M, Kumar V. *Introduction to Data Mining*. Boston: Addison-Wesley; 2005.
[9] Seidman S. *Network structure and minimum degree*. Social Networks. 1983; 5: 269–287.
[10] Singh D, Garg R. Ni-Louvain: *A novel algorithm to detect overlapping communities with influence analysis*. Journal of King Saud University - Computer and Information Sciences. 2021; 34.
[11] Newman M, Girvan M. *Finding and evaluating community structure in networks*. Physical Review E. 2004; 69: 026113.
[12] Newman M. *Analysis of weighted networks*. Physical Review E. 2004; 70: 056131.
[13] Fortunato S, Hric D. *Community detection in networks: A user guide*. Physics Reports. 2016; 659: 1–44.
[14] Newman MEJ. *Modularity and community structure in networks*. Proceedings of the National Academy of Sciences of the United States of America. 2006; 103: 8577–8582.