Comparative Analysis of Community Detection Algorithms

Janvi Patel

tuo93411@temple.edu

ABSTRACT

The detection of communities in complex networks has become a critical and highly relevant issue with numerous applications. Community Detection allows for a better understanding of the properties of dynamic processes occurring within a network. However, assessing the efficiency of various community detection algorithms in terms of accuracy and computational time remains an open question, especially in cases where there is no ground truth available for the underlying communities, which is often the case in real-world networks. Evaluating algorithms on real-world networks has limitations, such as the subjective definition of communities and the inability to control their properties. Furthermore, most community detection algorithms fail to adapt to changes in network topology, making it difficult to detect communities in evolving networks. In this study, I aim to address these issues by evaluating five state-of-the-art algorithms on a set of real-world and artificial BA networks and analyzing their performance under different scenarios. These findings will be a valuable contribution to assisting researchers in selecting appropriate algorithms for their own experiments using the igraph library.

INTRODUCTION

Community detection has become a crucial problem in network science with a range of applications. Firstly, it uncovers the internal organization of a network at a coarse-grained level, revealing hidden relationships between nodes that may not be easily observed through empirical testing. Secondly, it enhances our understanding of dynamic processes occurring within a network. For instance, the community structure of a graph significantly impacts the spread of epidemics and innovation, making it a crucial factor to consider.

Given its significance, it is not surprising that numerous community detection methods have emerged, drawing upon a range of disciplines such as applied mathematics, sociology, computer science, biology, and statistical physics. These methods aim to improve the identification of significant communities, while minimizing computational complexity. However, due to variations in the definitions of community employed by these algorithms, the results are not always directly comparable. Moreover, in many real-world applications, a ground truth for the identification of nodes to communities is unavailable, making it difficult to assess the reliability of community detection procedures. To mitigate these challenges and evaluate algorithm reliability, several benchmarks have been devised.

Why artificial Barabasi-Albert (BA) model?

Since the BA model generates scale-free networks with a power-law degree distribution, it resembles real-world networks, such as social networks and biological networks. Therefore, evaluating community detection algorithms on BA networks can provide insight into the algorithms' ability to identify communities in real-world networks.

Additionally, the BA model allows for the generation of networks with varying sizes and degrees of clustering, which enables the evaluation of community detection algorithms under different network topologies. This evaluation can provide valuable information on the robustness and scalability of community detection algorithms.

This study evaluates eight state-of-the-art community detection algorithms on both real networks and an artificial BA network for unweighted graphs with non-overlapping communities.

My contributions in this study are twofold:

1. I aim to determine which is the most suited algorithm in most circumstances based on observable properties of the network under consideration while detecting communities in real-world networks.
2. To investigate how communities are formed as the network evolves or undergoes changes in topology, and utilize different parameters as indicators to determine the reliability ranges of various algorithms

The methods section provides detailed information about the algorithms used in this study. Overall, this study contributes to the advancement of community detection in network science by evaluating the reliability and effectiveness of various algorithms.

RELATED WORK

The study [12] compares the accuracy and computing time of eight state-of-the-art community detection algorithms using the Lancichinetti-Fortunato-Radicchi benchmark graph, which allows for an objective definition of communities. The authors provide guidelines for choosing the most suitable algorithm for a given network based on observable properties and highlight the limitations of specific algorithms based on macroscopic network properties. The study also examines the dependency on network size for both predicting power and computing time.

Orman and Labatut [17] compared five community detection algorithms to study the behavior of mixing parameters and evaluate the performance of community detection algorithms concluding that walktrap and multilevel methods perform better. Lancichinetti and Fortunato [13] compared the performance of 12 different clustering algorithms using the GN and the LFR benchmarks. The authors focus on mixing parameters to perform comparative analysis for different values of network sizes

In reference [19], the effects of network structural and topological characteristics on the performance of clustering algorithms were examined. The authors specifically analyzed network size, number and size of communities, average node connectivity, and inter-intra cluster edge ratio (mixing). Their major contribution was the systematic evaluation of community detection algorithms across diverse network topologies. The findings indicate that the topology of a network has a substantial impact on the outcomes of community detection, indicating the necessity for the improvement of community detection algorithms.

Although there are several research papers that discuss and compare community detection algorithms using distinct methodologies [14], [15], [16], very few of them have explored the use of the growth parameter as a means of analyzing the formation of communities as a network evolves. Additionally, a few studies have tested these algorithms on the artificially generated Barabasi-Albert (BA) model. Therefore, my goal is to investigate the formation of communities during network evolution and how different community detection algorithms detect these communities.

METHODOLOGY

This methodology section outlines the details of the evaluated algorithms, evaluation metrics, and datasets used in this study. Figure 1 illustrates the methodology.

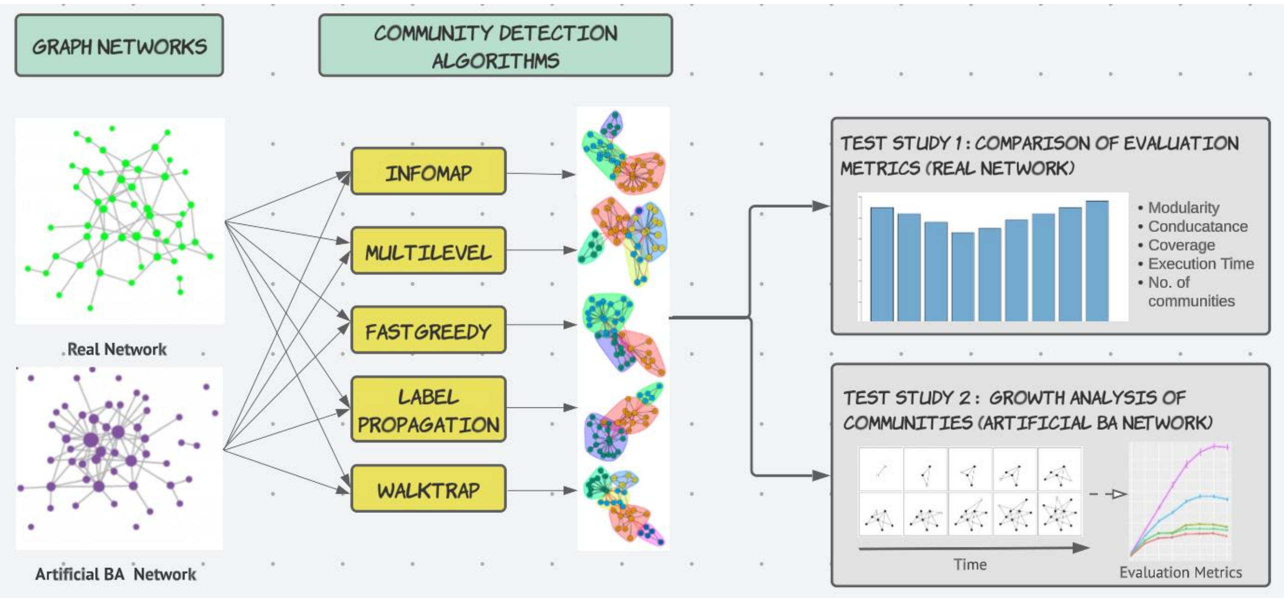

Figure 1. Methodology

A. The Evaluated Algorithms

Numerous community detection methods have been proposed in the past few years. These methods allow researchers to reveal communities in networks that can be used in a wide range of applications i.e., recommendation systems, innovation diffusion, viral marketing, etc. F. Berardo de Sousa and L. [1] provided a comparison of Random Walks, Infomap, Label Propagation, Spin Glass and Multi-Level algorithms using modularity and running time metrics. In this study, I have used Python's igraph [2] library to compare community detection algorithms.

The following five community detection algorithms were evaluated in this study:

- *Fast Greedy Modularity Optimization* - It is a greedy community detection algorithm proposed by Clauset et al [7] that optimizes the modularity score. To begin with, the algorithm assigns each node to a separate community, forming an initial non-clustered state. It then proceeds to calculate the expected improvement in modularity for each possible pair of communities, selects the pair that provides the highest increase in modularity, and combines them into a new community.

- *Label Propagation* - The algorithm was first presented by Raghavan et al. [8], and it operates on the assumption that each node in the network belongs to the same community as the majority of its neighboring nodes. To begin, the algorithm initializes a unique label (community) for each node in the network. The nodes are then randomly ordered, and the algorithm iterates through the sequence, assigning each node the label of the majority of its neighbors.

- *Infomap* - In 2008, Rosvall and Bergstrom introduced an algorithm [9] that transforms the task of identifying communities in networks into an optimization problem of compressing information about the dynamic process generated by a random walk within the network. This involves optimizing a quality function, which is the minimum description length of the random walk.

- *Multi-Level Modularity Optimization* - Blondel et al. introduced an algorithm [10] that employs a different approach to optimizing modularity compared to the Fastgreedy method. This algorithm initially assigns each node in the network to a distinct community, and then iteratively moves nodes to the community of one of its neighbors with which it produces the greatest positive contribution to modularity. This process is repeated for all nodes until no further improvement can be made.

- *Walktrap* - Pon & Latapy developed an algorithm [11] that utilizes hierarchical clustering. The algorithm is based on the notion that short-distance random walks tend to remain within the same community. Initially, the algorithm begins with a non-clustered partition and then computes the distances between all adjacent nodes. Next, the algorithm selects two neighboring communities, merges them into a new community, and updates the distances between communities.

B. Evaluation Metrics

The following evaluation metrics were used to compare and evaluate the performance of community detection algorithms:

- *Modularity* - a metric that evaluates the quality of a community structure by quantifying the difference between the actual number of edges within communities and the expected number of edges in a random network with the same degree distribution. A higher modularity score indicates a better community structure, with a greater number of edges within communities than expected by chance.

- *Conductance* - a metric that quantifies the extent to which a community is internally connected and externally isolated from the rest of the network. Communities with lower conductance scores are considered more tightly knit and cohesive, with a stronger internal connection and a weaker connection with the rest of the network.

- *Coverage* - a metric that quantifies the proportion of nodes in the network assigned to a community. Higher coverage indicates that more nodes have been assigned to communities, which can be beneficial for downstream analyses that rely on community membership information.

- *Execution time* - a critical evaluation factor that measures the time required for a community detection algorithm to complete its task, typically measured in seconds or minutes.

Dataset	Nodes	Edges
Netscience	1589	2742
Facebook	3892	17262
Artificial BA	2000	3999

Table I. Datasets for analysis

C. Data Sets & Experimental Setup

To evaluate the performance of the community detection algorithms, two sets of databases were utilized: Real networks and Artificial BA networks. Table I displays the size of both networks. These datasets are described in detail below.

- ***Real Networks***

To conduct complementary analysis, two actual networks were selected. The first dataset used for basic testing is Netscience [3], which consists of a co-authorship network of scientists specializing in network theory. For intermediate analysis, a slightly larger network from Facebook [4] was chosen. This network consists of verified Facebook pages for TV shows, where nodes represent pages and edges represent mutual likes.

- ***Artificial BA Network***

I utilized the NetworkX Barabási-Albert model to construct a scale-free network and investigated the effect of network growth on the formation of communities. Specifically, I used the Growth and Preferential Attachment methods to generate two sets of networks with different edge probability values, m=2 and m=4. By generating artificial scale-free networks, I aimed to provide a more controlled and structured environment to study the behavior of community detection algorithms.The subgraphs for analysis were created using the logspace function in numpy. This function generated evenly spaced numbers on a logarithmic scale between 10^1 and 10^log10(n). The resulting array contained 10 integer values that were used to define the sizes of the subgraphs to be analyzed. This approach allowed for the analysis of how the structure of the network changes as it grows, with subgraph sizes increasing logarithmically with the total number of nodes. The choice of subgraph sizes was made carefully to ensure meaningful results.

As discussed in the introduction section, creating artificial networks using the BA model has several benefits, including the ability to generate networks with varying topologies and controlled community structures. These networks can be used as a benchmark for evaluating the performance of community detection algorithms, enabling researchers to develop and refine these algorithms to better identify communities in real-world networks.I also assessed each algorithm's accuracy by comparing the actual number of communities in the network to the expected number and analyzing other relevant metrics.

RESULTS

The results section presents a comparative analysis of different community detection algorithms' accuracy and computing time using various evaluation metrics, as outlined in the methods section. The assessment is conducted on two datasets to evaluate the algorithms performance under different network structures. The section is organized as follows:

- ***Results for Real Network***

In the first part of the results section, I compared the accuracy and computing time of various community detection algorithms in various real networks. The findings reveal that the accuracy of the algorithms decreases as the complexity of the network increases. As shown in Figure 2 & Figure 3 the number of communities detected by different algorithms did not give any clear indication of which algorithm was better.

However, in terms of modularity, the multilevel algorithm followed by label propagation and fast greedy algorithms gave the highest modularity scores, indicating that these algorithms were able to find distinct communities in the network. Surprisingly, the walktrap algorithm performed better on the Netscience network, giving a high modularity score. However, it performed poorly on the Facebook network, indicating that the algorithm may not be suitable for all types of networks.

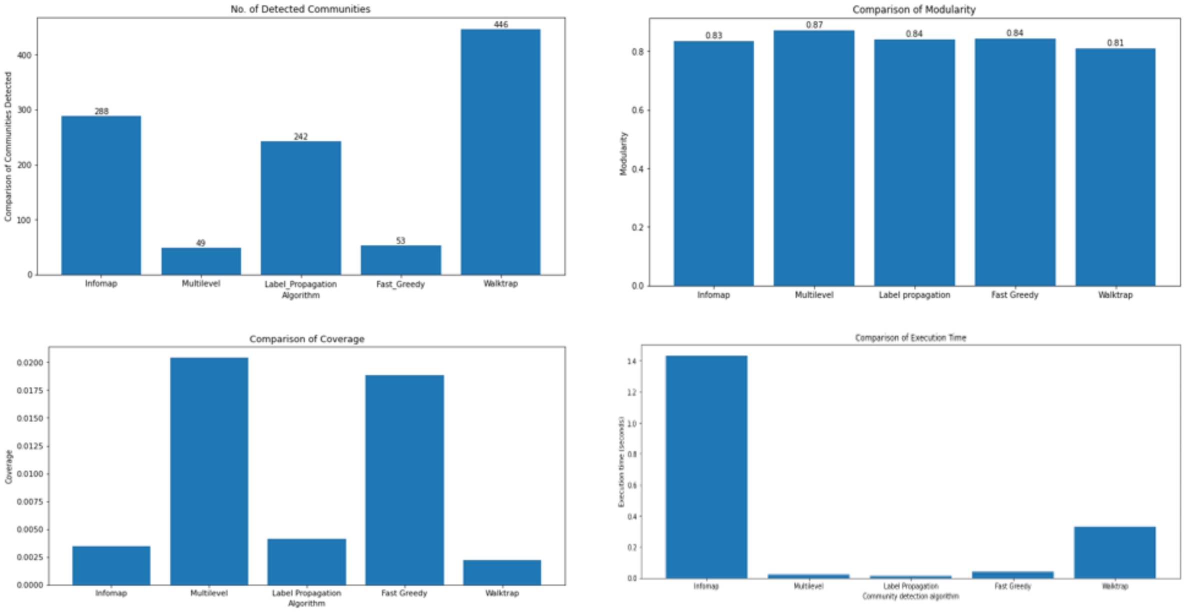

Figure 2. Comparison of different Metrics (Facebook Network)

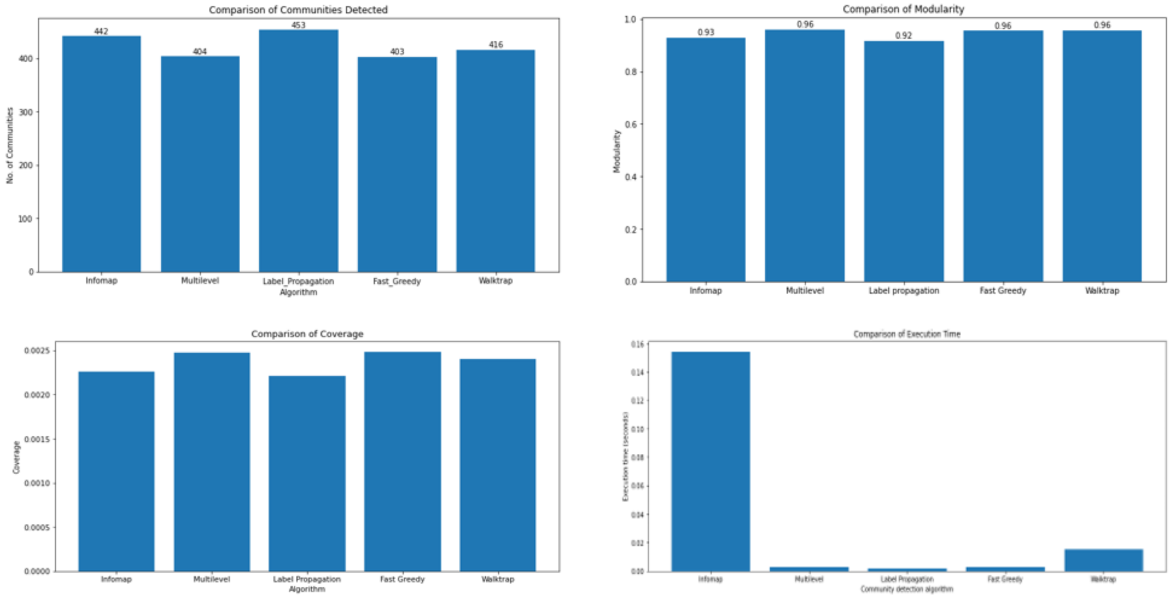

Figure 3. Comparison of different Metrics (Netscience Network)

Regarding network coverage, it can be observed that both the multilevel and fastgreedy algorithms outperformed the other algorithms in both networks. These algorithms demonstrated superior performance in terms of assigning nodes to the correct communities compared to the other methods tested. In terms of execution time excluding the infomap all other algorithms were able to detect communities in a lesser amount of time. The Infomap's time-consuming nature is due to its optimization-based approach, iterative nature, stochastic exploration, and hierarchical structure.

❖ *Results for Artificial BA Network*

In the second part of the results section, I compared the accuracy and computing time of various community detection algorithms in an artificial Barabási-Albert (BA) network. We investigate the performance of these algorithms under different network topologies and growth rates to determine the optimal algorithm for community detection in such networks. I have used various evaluation metrics outlined in the method section to assess the accuracy of community detection algorithms.

The result of increasing the number of nodes and edges in a network is that the number of communities also tends to increase [Figure 4]. This is because as the network becomes more complex with more nodes and edges, some nodes will become more important than others and will have many connections (i.e., hubs), while others will have few connections and be poorly connected.

Fastgreedy and Multilevel algorithms have proven to be efficient in identifying hubs within a network, as they can effectively detect nodes that have many connections with other nodes. In contrast, walktrap and infomap algorithms have shown to have lower modularity and higher numbers of communities. This is because these algorithms tend to identify many isolated nodes within the network, which can lead to an overall lower level of connectivity and fewer hubs. The label propagation algorithm performed the worst in this regard, as it was unable to distinguish between different communities within the network and instead identified the entire network as a whole. [Figure 4].

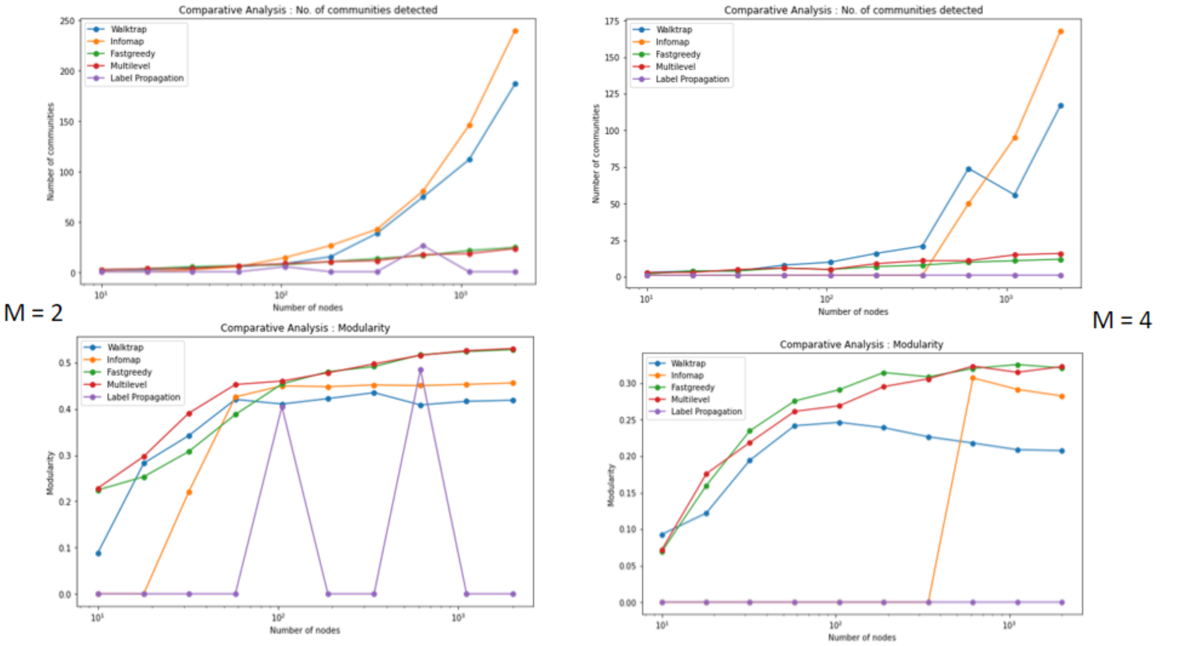

Figure 4. Comparison of Modularity and Coverage (Artificial BA Network)

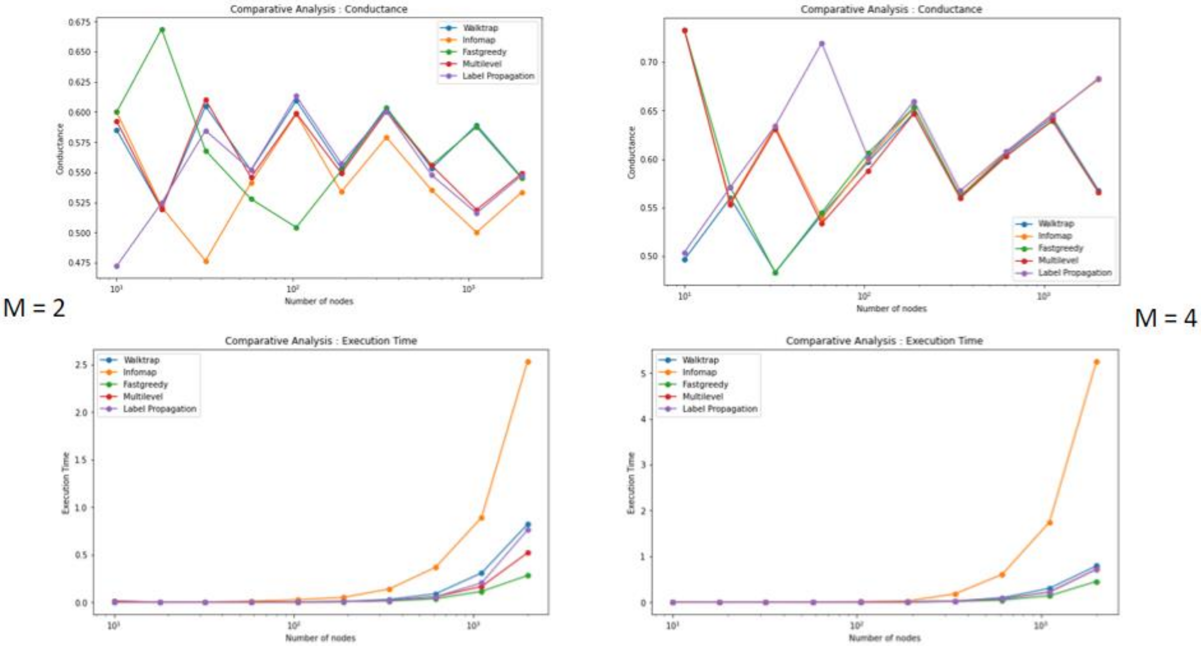

Figure 5. Comparison of Conductance and Execution Time (Artificial BA Network)

As shown in Figure 5 during the initial phase of network growth, when the number of nodes is limited, the conductance of communities tends to be high. This is because the communities are closely interconnected, and there are relatively fewer edges connecting them to the rest of the network. However, as more nodes and edges are added to the network, the conductance of communities decreases.

Overall, the results provide insights into the performance of different community detection algorithms in artificial BA networks and highlight the importance of choosing the appropriate algorithm for a particular network structure.

DISCUSSION

In the discussion section of this study, the results of a comparative analysis of five community detection algorithms available in the "igraph" package were presented. The purpose of the analysis was to assess the accuracy and computing time of these algorithms on both real-world and artificially generated graphs. The study found that the Multilevel, Walktrap, and Fastgreedy algorithms outperformed the others in terms of accuracy. The modularity of a network was found to be a significant factor in the accuracy of community detection algorithms, as a higher modularity indicates a stronger division of the network into distinct communities.

Moreover, the study also highlighted the impact of the value of "m" on the number of communities and network connectivity. A higher value of "m" resulted in more connections with fewer communities, while a lower value of "m" led to a sparse network with more distinct communities. These findings have important implications for researchers who are looking to use community detection algorithms for their own experiments using the igraph library. They provide valuable insights into the performance of different algorithms and factors that influence their accuracy, which can help researchers make informed decisions when selecting or discarding algorithms.

For the future scope of this study, there are several avenues to explore to further validate and improve the proposed approach for analyzing scale-free networks. One potential area of research is to apply the proposed approach to analyze and detect communities in random networks. Random networks differ from scale-free networks in terms of their topology, and therefore, evaluating the effectiveness of the proposed approach on these networks would provide a more comprehensive understanding of their performance.

Overall, further research in these areas would help refine and validate the proposed approach for community detection in scale-free networks, ultimately leading to improved methods for analyzing complex real-world networks.

ACKNOWLEDGEMENTS

I would take this opportunity to express my gratitude to those who have been instrumental in the successful completion of this project. I want to express my gratitude to Professor Zoran Obradovic and our Teaching Assistant, Dr. Jumanah Alshehri, for their exceptional assistance and guidance throughout the project and for giving me the opportunity and the environment to learn and grow.

REFERENCES

1. F. Berardo de Sousa and L. Zhao, "Evaluating and Comparing the IGraph Community Detection Algorithms," 2014 Brazilian Conference on Intelligent Systems, Sao Paulo, Brazil, 2014, pp. 408-413, doi: 10.1109/BRACIS.2014.79.
2. Csárdi, Gábor and Tamás Nepusz. “The igraph software package for complex network research.” (2006).
3. http://www-personal.umich.edu/~mejn/netdata/
4. https://github.com/benedekrozemberczki/datasets
5. Wikipedia contributors. “Barabási–Albert Model.” Wikipedia, Feb. 2023, en.wikipedia.org/wiki/Barab%C3%A1si%E2%80%93Albert_model.
6. Linhares, C.D.G., Ponciano, J.R., Pereira, F.S.F. et al. Visual analysis for evaluation of community detection algorithms. Multimed Tools Appl 79, 17645–17667 (2020). https://doi.org/10.1007/s11042-020-08700-4
7. Clauset, A., Newman, M. E. & Moore, C. Finding community structure in very large networks. Physical Review E 70, 066111 (2004).
8. Raghavan, U. N., Albert, R. & Kumara, S. Near linear time algorithm to detect community structures in large-scale networks. Physical Review E 76, 036106 (2007).
9. Rosvall, M. & Bergstrom, C. T. An information-theoretic framework for resolving community structure in complex networks. Proceedings of the National Academy of Sciences 104, 7327–7331 (2007).
10. Blondel, V. D., Guillaume, J.-L., Lambiotte, R. & Lefebvre, E. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment 2008, P10008 (2008).
11. Pons, P. & Latapy, M. Computing communities in large networks using random walks. In Computer and Information Sciences-ISCIS 2005, 284–293 (Springer, 2005).
12. Yang, Z., Algesheimer, R. & Tessone, C. A Comparative Analysis of Community Detection Algorithms on Artificial Networks. Sci Rep 6, 30750 (2016). https://doi.org/10.1038/srep30750
13. A. Lancichinetti and S. Fortunato, “Community detection algorithms: A comparative analysis,” Physical Review E, vol. 80, p. 056117, 2009.
14. S. Fortunato, “Community detection in graphs,” Physics Reports, vol. 486, no. 3-5, pp. 75–174, 2010.
15. J. Leskovec, K. Lang, and M. Mahoney, “Empirical comparison of algorithms for network community detection,” in ACM WWW International Conference on World Wide Web (WWW), 2010, pp. 1–11.
16. G. Orman, V. Labatut, and H. Cherifi, “Qualitative comparison of community detection algorithms,” in Digital Information and Communication Technology and Its Applications, ser. Lecture Notes in Computer Science.
17. Orman, G. K. & Labatut, V. A comparison of community detection algorithms on artificial networks. In Discovery Science 242–256 (Springer, 2009).
18. https://www.tandfonline.com/doi/pdf/10.1080/24751839.2019.1686683

19. M. Q. Pasta and F. Zaidi, "Topology of Complex Networks and Performance Limitations of Community Detection Algorithms," in IEEE Access, vol. 5, pp. 10901-10914, 2017, doi: 10.1109/ACCESS.2017.2714018.