Modularity based Community Detection in Social Networks

Shyam Sundar Meena, Vrinda Tokekar

Abstract: The community detection is an interesting and highly focused area in the analysis of complex networks (CNA). It identifies closely connected clusters of nodes. In recent years, several approaches have been proposed for community detection and validation of the result. Community detection approaches that use modularity as a measure have given much weight-age by the research community. Various modularity based community detection approaches are given for different domains. The network in the real-world may be weighted, heterogeneous or dynamic. So, it is inappropriate to apply the same algorithm for all types of networks because it may generate incorrect result. Here, literature in the area of community detection and the result evaluation has been extended with an aim to identify various shortcomings. We think that the contribution of facts given in this paper can be very useful for further research.

Keywords: Community detection, Networks, Modularity, NMI.

I. INTRODUCTION

From the last decade, research in the area of network analysis is growing day by day. Social network describes the connection between individuals. It is generally modeled through graph. Some examples of such type of networks are biological, road, social network etc. Community structure is an interesting property that is followed by networks of real-world. A community is a group of highly similar nodes [1]. In Figure 1, a network with three communities is shown, in which nodes within a community are closely related to each other and much less related to nodes outside the community. Community Detection task in graph is alike to clustering of data. It is called unsupervised learning because the number of communities in networks is not known in advance. Community detection can be performed in three ways: using topological features or using various details associated with nodes and edges or by applying a combination of both. Graph representation of real-world network may be very complex. We need multiple approaches to identify communities in networks, depending on the suitability of the networks.

Community detection can be used to categorize entities in the groups. For example to identify the relevant structure in the protein-protein interaction network, to find the similar category web-pages in network of World Wide Web, to detect the related papers in citation networks and a group of person in social networks [2].
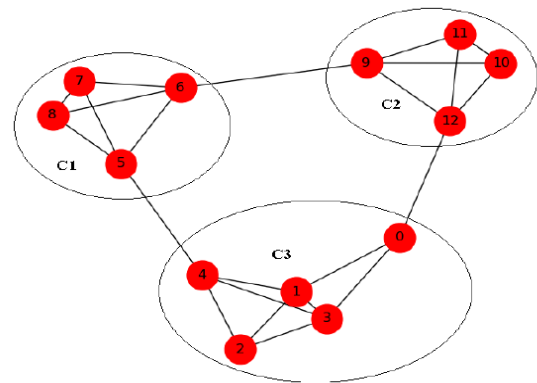

Fig. 1. Community Structure

In the past decade, modularity optimization-based techniques have been proposed by various researchers. The main focus was on the maximization of modularity [1], [3] - [4]. Modularity optimization approaches include spectral methods [3], extreme optimization [5], greedy algorithms [6], sampling method [7] simulated annealing [8], and mathematical programming [9].

Modularity is one of the indicators of the network structure or graph. It was designed to measure the dissociation power of a network in a module. There are narrow connections between nodes in a module in a high-modulus network, but there are rare connections between nodes in different modules. It is used by researchers for measuring the strength of the result of community detection algorithms. However, maximizing modularity has two problems. Firstly, if the resolution parameter value is low, larger communities will be formed by merging smaller communities. Second, it divides large communities into multiple smaller communities when the resolution is high. These trends are contrary to the network with big changes in the size of the community [10], [11]. In other cases, large communities are created by the merger of communities. This phenomenon depends on the number of edges and the degree of interconnection between communities. This issue is also termed resolution limit problem [11] [12].

The main focus of this paper is to study community identification in real-world networks and to understand the issues in real word networks analysis.

*Retrieval Number: B3382129219/2020©BEIESP
DOI: 10.35940/ijeat.B3382.029320
Journal Website: www.ijeat.org*

564

*Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication*

II. COMMUNITY DETECTION

The main issue behind analysis of social network is the investigation of community [3]. Much more has been researched for providing comprehensive analysis of networks. In this section, we survey the current literature on the community identification problems. The Newman-Girvan modularity [1], primarily introduced to determine the stopping condition for the Girvan and Newman algorithms. It has got high attention from the researchers. Modularity is an important and well know function for quality measurement in literature. This was one of the very first efforts that have been done to understand the clustering theory. It includes all the main components and questions, from defining a community to choosing a null model, to signify "Strength" of partitions.

As per the concept of various proposed works in literature, high value of modularity specifies goodness of partitions. Thus, if some division of the graph has the highest value of it then the corresponding division is called best. This is a key motive for maximizing modularity. Fully optimization of modularity is impossible because a graph can be partitioned in enormous ways. As per the community of researchers, Modularity maximization is an NP-complete problem [13]. However, in recent years various heuristic-based approaches have been given in the literature. The main aim of these approaches is to achieve a high value of modularity.

A greedy method based algorithm was given by Newman [1] in 2004. It is the first algorithm that mainly focused on the maximization of modularity. It is based on agglomerative hierarchical clustering. It merges nodes successively to make bigger communities. Such that at the end of this process the value of modularity should increase. The count of groups decreases from N to $N - 1$ when we add a new edge between disconnected nodes and a newly partitioned graph is created. The selection of edge is done such that finally, it gives the highest rise in the value of modularity as compared to the previous configuration. In the same way, all other edges are also added. If the earlier partition is not modified by adding an edge, the modularity remains unchanged. It identifies N partitions of network that are numbered from N to 1. In each step modularity is calculated, it helps in the selection of best merger of communities.

As Newman and Girvan [1] point out, the biggest drawback of its algorithm is heavy computational requirements. In the network of m edges and n vertices, $O(m^2 n)$ time is required in the worst case and when the network is scattered, the worst-case time changes to $O(n^3)$. Newman [6] gave a new algorithm to identify the communities. The algorithm works on different principles from the Girvan and Newman [1]. But the results produced by this are qualitatively same. The computation time taken by it in the worst case is $O((m + n) n)$. Algorithm of Newman [6] produces faster results as compared to the Newman and Girvan [1]. Authors in [14] indicated that in the case of a sparse network, each step of updating of adjacent matrix takes a large number of unwanted operations. They suggested data structures to perform the update operation in an effective manner. The algorithm of Clauset et al. [14] mainly works on the principle of the greedy method. It is not applicable for networks having greater than 500,000 nodes as noticed by Wakita and Tsurumi [15]. They found that this computational

ineffectiveness comes due to the unbalanced association of communities. In order to balance community merging, the authors in [15] presented three types of factors for measuring the balance of pairs of communities. To carry out the job of uniting groups without discrimination, it uses modularity. This change makes the algorithm scalable up to 10,000,000 nodes.

Blondel et al. [16] developed a greedy based approach for the optimization of modularity. It is a very different algorithm than the one discussed in the above section. It is also called the Louvain algorithm. In the starting, it assumes that each node of given network belongs to a separate community. The algorithm then iterates in random order for all nodes. For each node, it computes a change in modularity when a node moves in other communities from its current community. Finally, the node is moved to the neighboring community, which gives the greatest change in modularity. If none of the possible movements leads to an increase in modularity, the node remains in its present community. The first phase ends when modularity does not increase. When it reached in the final state, we found a partition that gives local maximum modularity. The next phase applies coarsening. Each community found in the first phase is now converted into a new node. Edges within the old community are attached as a self-loop to the new node. By summing the weights of all substituted edges the weight of this self-loop is determined. In the original graph, all the edges between the two communities are replaced by new edges between the respective nodes. Weight of new edges is calculated by summing the weight of replaced edges. These two steps are repeated until the first stage can provide an increase in modularity. The algorithm takes very small iteration in the process of community detection. It spends most of computation time in first step. Nonetheless, the Louvain algorithm findings are influenced by the selection of nodes in the first step [17].

The principal characteristics of method proposed by authors in [18] for discovering the internal structure of the community in the network are: i) it is focuses on global information, ii) the cost of computation is low, iii) it is applicable for large network analysis, iv) it uses a new measure of centralization of boundaries to rank all the boundaries of the network and v) it is able to achieve reliable results. Authors [19] expand the Louvain algorithm by adding an uncoarsening step. It is also called Louvain+. This change makes the algorithm a complete multi-level method. The non-coarsening phase expands vertices. After that, it applies the refinement (optimization) procedure. Louvain's community detection algorithm uses coarsening phase. It does not include uncoarsening phase. If we focus on optimization, it is clear that the un-coarsening improve the quality of the result. The experimental result also proves that the expansion of coarsening is not increased the computational cost of the original Louvain algorithm.

A parallel version of the Louvain algorithm is introduced in [20]. It has a very low computational cost. A new hash-based efficient technique is used for the processing and storage of large complex and dynamic networks.

565

The proposed algorithm produces similar results to the original Louvain sequential algorithm. The authors in [21] first point out the drawbacks of the original Louvain algorithm, and propose a hybrid method. This proposed algorithm increases group detection efficiency. It is made by combining the Louvain algorithm with the LPA (Label Propagation Algorithm) [22].

In [23], authors have given a GPUs based approach for detecting community in the real-world graph. It uses the Louvain method and parallelization heuristics. The community detection version with graphics processor is able to achieve a total acceleration of 12 times compared to the implementation of parallel processor-based algorithm. Both approaches use the same heuristics. Another acceleration of Louvain algorithm given in [24] restricts the rules of internal search. It is helpful for reducing unpromising candidates in advance. The results on various sizes, weighted and unweighted networks show very good accelerations.

The [25] article presents a Louvain based method for multi-core systems. It is used for the detection of parallel communities. It is an adaptive and rapid method. It tries to find a compromise between the assignment of fine-grained thread and the coarse assignment of thread. To do this, it searches free cores and determines the optimal count of cores that must be used to calculate the modularity of the gain. The results show that the use of parallelization in community detection is helpful for achieving faster acceleration than previous methods. Thus, this method is good for detecting communities in multi-core systems.

Authors [26] presented new algorithm that uses Louvain as a base and applies a distributed memory concept for detecting a community. They also presented several heuristics. Inclusion heuristics have been identified as crucial for improving performance and scalability. This parallel implementation showed good accelerations compared to the base version for a wide range of real networks.

Authors in [27] present a parallel version of the Louvain method based on a GPU. When implementing heuristics using the CUDA platform for a version with a GPU, the experimental results show a calculation time 12 times faster, and a theoretical study indicates acceleration 31 times over the CPU based parallel implementation of the Louvain. Besides the findings, the authors stated that GPUs can be used more efficiently if the data are large enough, but this is not the only important factor. In the case of this algorithm, the hierarchy data is also important to achieve higher performances. Random Neighbor Louvain Algorithm of V.A. Traag [28] reduces calculation time of ΔQ (modularity change). In this algorithm, the selection of a neighbor node is done randomly and a value of ΔQ is calculated for it. Based on the value of ΔQ, it is decided that the node will move to neighbor community. It is clear from the result that with a slight drop in modularity, this new algorithm can substantially speed up the actual algorithm. The random selection generates erratic results because the algorithm cannot guarantee that the selected community of neighbor is the most acceptable to move a node. The authors in [29] propose a new upgraded variation of the Louvain algorithm. It called Random Self-adaptive Neighbor. Here, neighbors are selected randomly by utilizing the principle of the low probability of an event.

III. EVALUATION MEASURES

A. Modularity [3]:

Community detection is a process of dividing a network into modules (also known as classes or clusters). To calculate the strength of this division process, modularity is used as a metric. The high value of modularity indicates that the number of connections between nodes within a module is more than the connections of these nodes to nodes of other modules. Modularity is also used to detect community structure in optimization-based approaches. For a network G (V, E), which has two communities, calculation of modularity value can be given as follows:

$$Q = \frac{1}{4E} \sum_{ij} \left(M_{ij} - \frac{d_i d_j}{2E} \right)(l_i l_j + 1) \tag{1}$$

Where E represents number of edges, d_i and d_j are degrees of nodes, the adjacency matrix is given by M_{ij} and l_i will be 1 if vertex i belongs to first community or -1 if it belongs to group 2.

Regardless of the popularity of the maximization of modularity, there are some facts unknown about the quality and importance of its results in case of the unknown modular structure. Most of the previous works have focused on the development of new ways to detect partitions of networks, instead of marking them effective for practical situations. Modularity optimization methods are suffered from resolution limit problem [30]. Maximizing modularity usually merge small communities into large communities. Authors in [31] found more deficiencies in modularity. Generally, Modularity optimization approaches for community detection may produce various types of division of a network. This problem called a degeneracy problem. This problem mostly appears when a network has a modular structure. The main reason behind it is the use of weak evaluation measures for community detection. The size of the network and the number of divisions it contains are the most important factors that affect the maximum value of modularity. These shortcomings can lead to an incorrect estimate of the number of communities [32].

B. Purity [33]:

This index matches each cluster generated by the procedure of group detection with the ground-truth.

$$\text{Purity}(S, C) = \frac{1}{n} \sum_i \max_j | S_i \cap C_j | \tag{2}$$

Where, n is used to represent number of samples, $C = \{C_1, C2 \ldots Cc\}$ is set of clusters obtained by community detection method and $S = \{S_1, S_2 \ldots S_k\}$ is the ground-truth.

C. Rand index (RI) [34]:

The most important details community detection algorithms give is that which sample will groups together and which ones are not. This is a main motivation behind the definition of Rand index. Rand Index is used to calculate the correlation between the ground truth and the result of the group identification. It can be defined as follows:

$$RI(A, B) = \frac{m_{11} + m_{00}}{m_{11} + m_{01} + m_{10} + m_{00}} \qquad (3)$$

where, A is used to represent original community assignment and B is a result of a community detection algorithm , m_{11} denotes the number of pairs of nodes that are in the same communities in both A and B, m_{00} is the number of pairs of nodes that are placed in the different communities in both A and B, m_{10} represent number of pairs of nodes that belong to the same community in A but are in the different community in B and m_{01} indicates the number of pairs of nodes that are put in the different communities in A but are in the same community in B. In general, a quantity of agreement between two partitions can be defined by m_{11} and m_{00}, while m_{10} and m_{01} can be used for the representation of disagreements. The RI takes a value between 0 and 1.

D.Normalized mutual information (NMI, [35]):

In the area of cluster analysis, the use of the Normalized Mutual Information (NMI) is much more modern than the earlier measures. This tests the similarity between the original community assignment and the result of the group detection algorithm. It has been taken from information theory, defined as:

$$NMI(A, B) = \frac{2 * I(A, B)}{[H(A) + H(B)]} \qquad (4)$$

Where, A = class labels, B = cluster labels, H (.) = Entropy and $I(A, B)$ = Mutual Information between A and B.

IV. CLASSIFICATION SUMMARY

In this section, different community detection approaches have been compared based on some characteristics. The most famous approaches to explore the community are discussed in Table I. The comparison is difficult on this topic because each methodology is based on different metrics and structures. Our comparison is based on the following features:

A. Approach:

Here we are mainly considering the community detection approach that work based on the hierarchical grouping. The objective of building a hierarchical structure of communities is to allow an analysis of a network at different resolutions. Two representative methods in this direction are the divisive hierarchical grouping (top-down) and the agglomerated hierarchical grouping (bottom-up).

- *Agglomerative:* This is a "bottom-up" strategy. Initially, each node belongs to a separate community. As the approach proceeds up in the hierarchy, merging of pairs of communities is done.
- *Divisive:* This is a "top-down" method: in the beginning, it assumes that all nodes are in a single community. It split a big community into multiple, as proceed down the hierarchy.

B. Principal of Approach:

Here, a description of the technical theory or algorithm is given.

C. Type of Network:

Here, various networks are classified based on the weight of the edges. A network is called weighted if links between two nodes have some weight otherwise it is called unweighted.

D. Network Direction:

If edges in the network have direction then it is called directed network and if the direction of links is not considered then this is known as an undirected network.

E. Network Size:

The size of a network is generally defined by the number of nodes it has. Network size is an important parameter for testing a new community detection approach. Here, a classification of networks has done in three categories. These categories are defined as small, medium and large.

V. RESULT ANALYSIS

Here, Louvain [16] and Random Neighbor Louvain Algorithm of V.A. Traag [28] are implemented and compared. Both algorithms use modularity optimization for community detection. The code given by Thomas Aynaud [43] is used for further extension. Four datasets from [44] are used for analysis. The names of datasets are Dolphins, Football, PolBooks, and Karate. In the Table II, results of community detection on various standard datasets is given. It is clear from the result that modularity optimization-based algorithm is unable to identify the actual number of communities for all type of networks.

The following research problems have been identified in community detection.

A. Scalability and Quality

Most of the algorithms of community detection give efficient results when the size of the network is small. As a result, current algorithms are less successful in finding a community and some algorithms are not even able to find communities. There is an opportunity to develop scalable algorithms to identify the quality of communities in the real network.

B. Direct versus Undirected Network

For the undirected network, most current algorithms work well, and only some algorithms work for a direct network, but they do not achieve good results.

C. Un-weighted versus weighted Network

It has been observed that most of the work done for community detection focuses on unweighted networks. But the nature of networks in the real-world can be weighted.

567

Table- I: Comparison of community detection approaches

Principal used	References	Solution Approach	Network Type	Network Direction	Size of Network (Number of Nodes)
Global modularity optimization	[1], [6], [36]	Divisive	Un-weighted	Undirected	Medium
	[14]	Agglomeration	Un-weighted	Undirected	Large
Local modularity optimization	[5], [37]	Divisive	Un-weighted	Undirected	Medium
	[38]	Agglomeration	Weighted	Undirected	Medium
	[39]	Agglomeration	Un-weighted	Undirected	Large
	[40]	Divisive	Un-weighted	Undirected	Large
Hybrid algorithms	[16], [41]	Agglomeration	Weighted	Undirected	Large
	[42]	Agglomeration	Un-weighted	Undirected	Large

Table- II: Number of communities identified by Louvain and Random selection based algorithms

Name of Dataset	Actual Count of Communities	Number of Communities detected	
		Louvain [16]	Random selection based Louvain [28]
Dolphins	4	5	Generated different number of communities for each run.
Football	12	10	
PolBooks	3	4	
Karate	2	4	

VI. CONCLUSION

In this research paper, we extend the literature on modularity based community detection. The other directions we have also included are the evaluation measures. It also focuses on the main strengths and limitations of each approach. High modularity indicates the goodness of partitions. Thus, the best partition must have a maximum value of modularity for a given graph. Fully optimization of modularity is impossible because a graph can be partitioned in a huge number of ways. In addition, the true maximum is not possible, because the optimization of modularity is an NP-complete problem. Modularity based approaches also suffered from the Resolution Limit Problem. Various evaluation measures proposed in the literature are also not applicable to all types of networks. Further research is required to detect the actual number of divisions of networks.

REFERENCES

1. M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," Physical review E, vol. 69, no. 2, p. 026113, 2004.
2. S. Fortunato, "Community detection in graphs," Physics reports, vol. 486, no. 3-5, pp. 75–174, 2010.
3. M. E. Newman, "Modularity and community structure in networks," Proceedings of the national academy of sciences, vol. 103, no. 23, pp. 8577–8582, 2006.
4. M. E. J. Newman, "Analysis of weighted networks." Physical Review E, vol. 70, no. 5, p. 56131, 2004.
5. J. Duch and A. Arenas, "Community detection in complex networks using extremal optimization," Physical review E, vol. 72, no. 2, p. 027104, 2005.
6. M. E. Newman, "Fast algorithm for detecting community structure in networks," Physical review E, vol. 69, no. 6, p. 066133, 2004.
7. M. Sales-Pardo, R. Guimer`a, A. A. Moreira, and L. A. N. Amaral, "Extracting the hierarchical organization of complex systems," Proceedings of the National Academy of Sciences, vol. 104, no. 39, pp. 15 224–15 229, 2007.
8. A. Medus, G. Acua, and C. Dorso, "Detection of community structures in networks via global optimization," Physica A: Statistical Mechanics and its Applications, vol. 358, no. 2, pp. 593 – 604, 2005.
9. G. Agarwal and D. Kempe, "Modularity-maximizing graph communities via mathematical programming," European Physical Journal B, vol. 66, no. 3, pp. 409–418, 2008.
10. M. Chen, T. Nguyen, and B. K. Szymanski, "On measuring the quality of a network community structure," in 2013 International Conference on Social Computing. IEEE, 2013, pp. 122–127.
11. S. B. Chen M, Nguyen T, "A new metric for quality of network community structure," arXiv preprint arXiv:1507.04308, 2015.
12. S. Fortunato and M. Barthelemy, "Resolution limit in community detection," Proceedings of the National Academy of Sciences, vol. 104, no. 1, pp. 36–41, 2007.
13. U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hoefer, Z. Nikoloski, and D. Wagner, "On modularity clustering," IEEE transactions on knowledge and data engineering, vol. 20, no. 2, pp. 172–188, 2008.
14. A. Clauset, M. E. Newman, and C. Moore, "Finding community structure in very large networks," Physical review E, vol. 70, no. 6, p. 066111, 2004.
15. K. Wakita and T. Tsurumi, "Finding community structure in mega-scale social networks," in Proceedings of the 16th international conference on World Wide Web. ACM, 2007, pp. 1275–1276.
16. V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," Journal of statistical mechanics: theory and experiment, vol. 2008, no. 10, p. P10008, 2008.
17. T. Chakraborty, S. Srinivasan, N. Ganguly, S. Bhowmick, and A. Mukherjee, "Constant communities in complex networks," Scientific reports, vol. 3, p. 1825, 2013.
18. P. De Meo, E. Ferrara, G. Fiumara, and A. Provetti, "Generalized louvain method for community detection in large networks," in 2011 11th International Conference on Intelligent Systems Design and Applications. IEEE, 2011, pp. 88–93.
19. O. Gach and J.-K. Hao, "Improving the louvain algorithm for community detection with modularity maximization," in International Conference on Artificial Evolution (Evolution Artificielle). Springer, 2013, pp. 145-156.
20. X. Que, F. Checconi, F. Petrini, and J. A. Gunnels, "Scalable community detection with the louvain algorithm," in 2015 IEEE International Parallel and Distributed Processing Symposium. IEEE 2015, pp. 28-37.
21. B. Hu, W. Li, X. Huo, Y. Liang, M. Gao, and P. Pei, "Improving Louvain algorithm for community detection," in 2016 International Conference on Artificial Intelligence and Engineering Applications. Atlantis Press, 2016.
22. U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," Physical review E, vol. 76, no. 3, p. 036106, 2007.
23. R. Forster, "Louvain community detection with parallel heuristics on gpus," in 2016 IEEE 20th Jubilee International Conference on Intelligent Engineering Systems (INES). IEEE, 2016, pp. 227–232.
24. S. Ryu and D. Kim, "Quick community detection of big graph data using modified louvain algorithm," in 2016 IEEE 18th International Conference on High Performance Computing and Communications. IEEE, 2016, pp. 1442–1445.
25. M. Fazlali, E. Moradi, and H. T. Malazi, "Adaptive parallel Louvain community detection on a multicore platform," Microprocessors and Microsystems, vol. 54, pp. 26–34, 2017.

Modularity based Community Detection in Social Networks

26. S. Ghosh, M. Halappanavar, A. Tumeo, A. Kalyanaraman, H. Lu,D. Chavarria-Miranda, A. Khan, and A. Gebremedhin, "Distributed louvain algorithm for graph community detection," in 2018 IEEE International Parallel and Distributed Processing Symposium (IPDPS).IEEE, 2018, pp. 885–895.

27. R. Forster, "Parallel louvain community detection optimized for gpus," arXiv preprint arXiv:1805.10904, 2018.

28. V. A. Traag, "Faster unfolding of communities: Speeding up the Louvain algorithm," Physical Review E, vol. 92, no. 3, p. 032801, 2015.

29. Z. Zhang, P. Pu, D. Han, and M. Tang, "Self-adaptive louvain algorithm: Fast and stable community detection algorithm based on the principle of small probability event," Physica A: Statistical Mechanics and its Applications, vol. 506, pp. 975–986, 2018.

30. A. Lancichinetti and S. Fortunato, "Limits of modularity maximization in community detection," Physical review E, vol. 84, no. 6, p. 066122, 2011.

31. B. H. Good, Y.-A. De Montjoye, and A. Clauset, "Performance of modularity maximization in practical contexts," Physical Review E, vol. 81, no. 4, p. 046106, 2010.

32. A. Kehagias and L. Pitsoulis, "Bad communities with high modularity," The European Physical Journal B, vol. 86, no. 7, p. 330, 2013.

33. C. D. Manning, P. Raghavan, and H. Sch¨utze, Introduction to Information Retrieval. New York, NY, USA: Cambridge University Press, 2008.

34. W. M. Rand, "Objective criteria for the evaluation of clustering methods," Journal of the American Statistical Association, vol. 66, no. 336, pp. 846–850, 1971. [Online]. Available: http://www.jstor.org/stable/2284239

35. L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," Journal of Statistical Mechanics: Theory and Experiment, vol. 2005, no. 09, p. P09008, 2005.

36. B. Saoud and A. Moussaoui, "Node similarity and modularity for finding communities in networks," Physica A: Statistical Mechanics and its Applications, vol. 492, pp. 1958–1966, 2018.

37. Z. L¨u and W. Huang, "Iterated tabu search for identifying community structure in complex networks," Physical Review E, vol. 80, no. 2, p.026130, 2009.

38. P. Schuetz and A. Caflisch, "Efficient modularity optimization by multistep greedy algorithm and vertex mover refinement," Physical Review E, vol. 77, no. 4, p. 046112, 2008.

39. Q. Ye, B. Wu, and B. Wang, "Detecting communities in massive networks efficiently with flexible resolution," in The influence of technology on social network analysis and mining. Springer, 2013, pp. 373–392.

40. J. Xiang, Z.-Z. Wang, H.-J. Li, Y. Zhang, S. Chen, C.-C. Liu, J.-M. Li, and L.-J. Guo, "Comparing local modularity optimization for detecting communities in networks," International Journal of Modern Physics C, vol. 28, no. 06, p. 1750084, 2017.

41. O. Gach and J.-K. Hao, "A memetic algorithm for community detection in complex networks," in International Conference on Parallel Problem Solving from Nature. Springer, 2012, pp. 327–336.

42. X. Liu and T. Murata, "Advanced modularity-specialized label propagation algorithm for detecting communities in networks," Physica A: Statistical Mechanics and its Applications, vol. 389, no. 7, pp.1493–1500, 2010.

43. T. Aynaud. Louvain community detection. [Online]. Available: https://github.com/taynaud/python-louvain

44. L. Zh, "Datasets (Karate, Dolphins, Football, Polbooks, Polblogs)," 5 2017. [Online]. Available: https://doi.org/10.6084/m9.figshare.4993895.v1

AUTHORS PROFILE

Mr. Shyam Sundar Meena is a research scholar at Institute of Engineering & Technology, Devi Ahilya Vishwavidyalaya, Indore and currently working as an assistant professor, Deparment of Computer Science & Engineering, SVIIT- SVVV, Indore. He has obtained Masters of Engineering degree from Institute of Engineering & Technology, DAVV, Indore in Computer Engineering (with specialization in Software Engineering) and Bachelor of Engineering from Shri Govindram Seksaria Institute of Technology and Science, Indore. He has 12 years of experience in the field of teaching.

Dr. (Mrs.) Vrinda Tokekar is Professor and Head, Department of Information Technology in Institute of Engineering & Technology and Head IT Center in Devi Ahilya Vishwavidyalaya, Indore. She received her Bachelor of Engineering, (Electrical & Electronics) (Hons.) from Birla Institute of Technology & Science, Pilani (Raj.), Master of Engineering (Computer Engineering) from SGSITS, Indore and Ph.D. in Computer Engineering from Devi Ahilya University, Indore, India. She has 32 years of academic and teaching experience in Graduate and post graduate engineering programs. Her areas of research are Wireless and Ad-hoc Networks, Network Security and Network Analysis. She has guided many PhDs.