Biosc.Biotech.Res.Comm. Special Issue Vol 13 No 14 (2020) Pp-388-393

BBRC
Bioscience Biotechnology Research Communications

Performance Evaluation of Community Detection Algorithms in Social Networks Analysis

Prajakta Vispute[1*] and Shirish Sane[2]
[1, 2]*Department of Computer Engineering*
K. K. Wagh Institute of Engineering Education & Research, Nashik, India
affiliated to Savitribai Phule Pune University (SPPU), Pune, Maharashtra, India.

ABSTRACT

In social network analysis, community identification unveils properties shared by nodes like area of research, communication, common interest and many more. The evolving nature of social networks necessitates dynamic community detection methods. To handle the continuous change in data, improved community detection algorithms are introduced in various applications. To find communities in dynamic SNA, static community detection methods can be used to generate base communities, which then can be modified for dynamic data. This paper deals with selection of suitable algorithm for detection of communities from static data based on different performance parameters and thus could be used for efficient detection of dynamic communities.

KEY WORDS: COMMUNITY DETECTION, DYNAMIC COMMUNITY DETECTION, GRAPH MINING, NETWORK ANALYSIS.

INTRODUCTION

Various studies recently have emphasized on the social structure of individuals and their direct or indirect communication based on common relation of interests known as social networks [Lei Tang & Huan Liu, 2010] [Maryam pourkazemi et al., 2013] [Fortunato & Hric, 2016]. A group of people where each member is familiar with some division or other is known as a social network and the study of social networks to understand their structure and behaviour is known as Social Network Analysis. The data and information of social network is effectively represented by text, tables or graphs. By standards, a social network can be built up in an organization, educational institute, or among any group of people with the help of social interaction and building personal relationships. The module of networks studied comprises of a computer, biological, financial, medical, physical, and transportation networks and much more. The problem area includes the analysis of thoughtful structure of the networks, the development of such arrangements, and methodology data is transmitted inside the networks. There are various graphical representations used to show univariate data like pie charts, histograms, scatter plots, bar graphs, etc. The precise nature of the representation to be chosen depends on the data set and application. A significant and highly efficient way to visualize the social networks is 'graph'. Individuals or organizations communicating with each other are represented by nodes and communication between nodes is represented by edges. A Graph is denoted by G (V, E), where V denotes node and E denotes edges.

Some of the SNA [Cuvelier et al., 2012] tasks include centrality analysis, community detection and role analysis and outlier detection. One of the most prominent tasks used in many applications is Community detection [S. Fortunato, 2010]. The propensity of relating people of similar characteristics leading to the formation of groups are called communities. Community detection

ARTICLE INFORMATION

*Corresponding Author: psvispute.it@gmail.com
Received 17th Oct 2020 Accepted after revision 23rd Dec 2020
Print ISSN: 0974-6455 Online ISSN: 2321-4007 CODEN: BBRCBA

Thomson Reuters ISI Web of Science Clarivate Analytics USA and Crossref Indexed Journal

Clarivate Analytics
crossref

NAAS Journal Score 2020 (4.31)

Online Contents Available at: http//www.bbrc.in/
Doi: http://dx.doi.org/10.21786/bbrc/13.14/90

aims to segregate a cohesive group present in the network. Community detection [Michele Coscia et al., 2011] can be node centric, group centric, network centric or hierarchy centric. Depending upon the type of data and application, different types of community detection methods are used which include Hierarchical clustering, minimum cut method, modularity maximization and clique based methods. Communities can be advantageous in various fields such as searching a group of similar behaviour customers for marketing and recommendations, discovering protein interaction networks in biological networks, finding a common research area in collaboration networks, finding a set of students with common interests in an institutional network and so on.

In static approach, the network will be divided into different communities based on static graph. The relationship between nodes and edges will not change in static graph. The static clustering based on single time view of network does not depict the appropriate structure of social network and unable to cover all the necessary characteristics of a network. However, static graphs cannot represent the continuous change in real data. To handle the continuous change in edges and nodes, dynamic community detection methods are useful. A dynamic community detection method captures the ongoing change in node positions due to interactions between them in the network and updates the communities accordingly. At present this is one of the most noteworthy research topics in the field if SNA. Based on current work, in the field of dynamic community detection algorithm, more emphasis is given on processing the incremental data effectively. The first step of this class of algorithm detects communities and in a second step, updates the communities for new data. To detect communities in first step existing static community detection methods can be used. These detected communities derived from the first step will be updated in a second step to handle the incremental data.

There are a number of classic static community detection algorithms are available. Use of efficient static community detection algorithm in first step provides a good platform to process incremental data effectively. This study focuses on three classic static community detection methods and comparison between them based on different parameters. It is useful to recommend appropriate community detection algorithm for the first step of dynamic community detection. The rest of the paper is organized as follows: Section II deals with details of existing algorithms while datasets, parameters and details of experimentation is provided in section III. Section IV deals with the analysis of experimental results. The paper concludes with Section V and Section VI provides conclusions and future scope.

Related Work: Over the period of time relation between nodes and edges doesn't change in static community detection methods. However, in evolving networks, change in the data may change the community structure. A dynamic network captures change in node positions depending on the ongoing change of interactions in the network and updates the communities accordingly. It is very important to update the identified communities to increase its usefulness. One of the methods used to deal with dynamic community detection is called incremental graph mining. In incremental graph mining algorithm [Z. Zhao et al., 2019] [Javadi Saeed et al., 2018] [Mohammad Ali Tabarzad et al., 2018], graph is constructed using available data and communities are discovered in an initial stage. As the new data arrives, it was processed and updated in the graph without reconstructing the graph already constructed. Community detection methods (either static or dynamic) [Lancichinetti et al., 2009] [Cazabet & Rossetti, 2019] can fetch valuable information about the social structure. The rest of the section deals with three well known hierarchical community detection techniques for community detection using static data called Girvan Newman [Girvan & Newman, 2002], Blondel (usually referred as Louvain) [Blondel et al., 2008] algorithm and Label propagation algorithm [Raghavan et al., 2007].

Girvan Newman algorithm: Social network analysis [Girvan & Newman, 2002] uses variations of hierarchical clustering methods. It could be divisive or agglomerative. In divisive hierarchical clustering, initially complete network is considered and edges are deleted until the communities are formed. On the other hand agglomerative clustering algorithms, considers every node as a cluster. And merge the pair of clusters until all clusters merged into a single cluster. The Girvan-Newman algorithm is an example of divisive algorithm. This algorithm uses the concept of 'edge Betweenness'. The number of shortest paths passing through the edge defines the edge betweenness. Girvan Newman algorithm detects communities by progressively removing edge(s) with the highest edge betweenness. The algorithm begins with a single node, calculates edge weights for paths going through that node, and then repeats it for every node in the graph and sums the weights for every edge. To calculate edge betweenness, all shortest paths in the graph are computed. The Girvan Newman algorithm generates non-overlapping communities.

Louvain algorithm: The Louvain algorithm [Blondel et al., 2008] is agglomerative clustering algorithm and known as a greedy optimization method. It works in two steps, initially it checks small communities by local modularity optimization. The nodes belonging to same community are aggregated in a second step. It recursively merges communities into a single node until maximum modularity is attained. The modularity enumerates the quality of an assignment of nodes to communities. Louvain algorithm generates non-overlapping communities.

Label propagation: Label propagation algorithm [Raghavan et al., 2007] [Garza and Schaeffer, 2019] is also known as a localized community detection algorithm. Initially, a unique label is assigned to each vertex. Iteratively, each vertex obtains a label with the most recurrences in the neighbouring vertices. Finally, vertices with same labels are grouped as communities. The Label

propagation algorithm generates non-overlapping communities.

Datasets, Parameters & Experimentation: Four datasets, applied to the community detection algorithm named Enron dataset, DBLP dataset, Zachary's Karate club dataset and Facebook dataset. Initially datasets are processed to generate the information in the required form. Unnecessary data will be removed and datasets of required size depending upon the specified parameters are generated from this research. The data file is it iterated through each row, examining node1 and node 2 columns to create a pairwise combination of (u, v). As node 1 is connected to node 2, update the edge list and accordingly and an undirected graph is generated.

A. Enron email Dataset: Large set of email messages, the Enron email data contains email communications between around 150 employees of Enron Corporation from 1999 to 2002. For this work, emails exchanged for the year 2001 are considered. As existing dynamic community detection algorithms fetch the data month wise, January 2001 data will generate. It consists of employees (nodes), email communication between the employees (edges) and number of times two employees communicate with each other (edge weight). The dataset will be given as an input to classic static algorithms to find out communities and related parameters.

B. DBLP Dataset: DBLP stands for digital bibliography and library project. It provides a comprehensive list of research papers from various fields in computer science published over the years. It contains information of approx 1,632,442 research papers. It is also known as co-authorship network where two authors are connected if they publish at least one paper together. Common publication between two authors will be represented by an edge. Multiple edges between two nodes represents pairs of authors have written multiple publications together. This data set defines ground-truth community. The DBLP dataset has a massive list of publications. Data for the year 2000 is considered to generate the dataset. It consists of 357 Authors (nodes), 454 common publication details (edges) and number of times two authors publish paper together (edge weight).

C. Zachary's Karate Club dataset: Wayne Zachary collected the data from the members of a university karate club in 1977. It is widely used dataset in community detection research field. In the graph, member of the club is represented by a node and the communication between the two members of the club is represented by an edge. The data set considered here contains 34 members of the club and 78 ties between the members of the club. No multiple edges are present in this dataset.

D. Facebook Dataset: This dataset consists of friends lists from Facebook. This dataset shows the relationship between facebook users. Here interaction between 178 users (nodes) with 267 communications (edges) is taken into consideration. Parameters are calculated to understand the statistics of the data represented in the graph. The values related to the basic measures related to the datasets used in this research are as shown in table 1.

Table 1. Dataset Characteristics:

Dataset	# Nodes	# Edges	Average Degree	Graph Density
Enron	146	728	6.192	0.043
DBLP	357	454	2.543	0.007
Karate Club	34	78	4.588	0.139
Facebook	178	267	3	0.017

In case of non-overlapping community detection, network divides into groups of nodes with dense connections on the inside and sparser connections between groups. To understand the structure of data, degree distribution of the data set is produced. The degree distribution provides the number of nodes in the graph at each degree. The degree distribution d (k) of a graph or network is the fraction of nodes with degree k. So if there are n nodes in total in a network and q (k) of them have degree k, then d(k) = q(k)/n. Degree distribution of used dataset is shown in figure 1 to 4.

Figure 1: Degree Distribution Enron Dataset

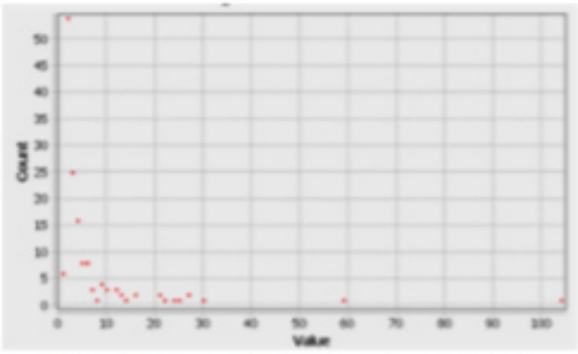
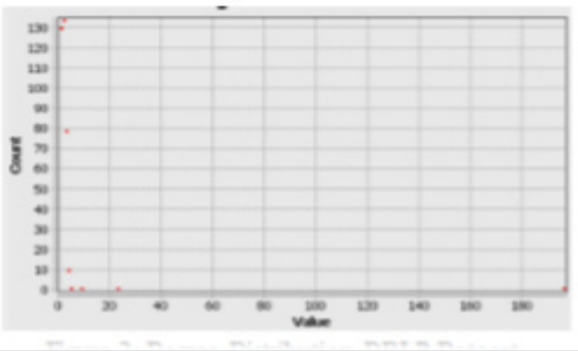


Figure 2: Degree Distribution DBLP Dataset



Degree distribution of Enron email dataset with 728 sent-emails between 146 emplyoees is shown in figure 1, DBLP dataset with 454 common publication details between 357 authors is shown in figure 2, Karate club data set with 34 members of the club and 78 ties between the members of the club is shown in figure 3, Facebook

dataset with interaction between 178 users with 267 communications is shown in figure 4.

Figure 3: Degree Distribution Karate Club Dataset

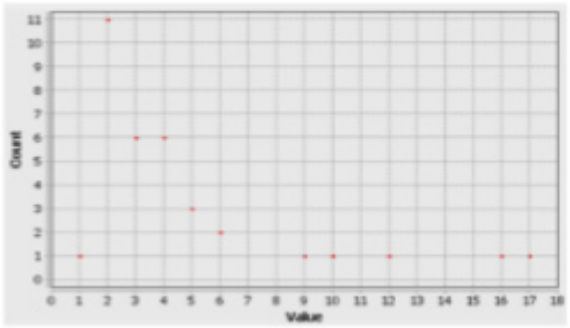
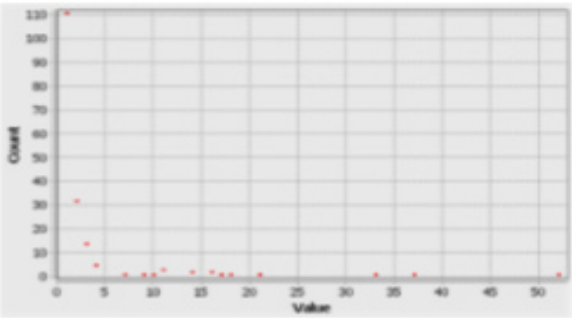


Figure 4: Degree Distribution facebook Dataset



In this research, non-overlapping algorithms like Girvan Newman, Louvain and LPA are used to detect the communities. These algorithms are applied to data sets from different domain to a range of degree distribution. Extracted communities are evaluated using different parameters. Basically a graph consists of vertices and edges. The size of the graph is the number of vertices in V, degree of a vertex is the number edge attached to it. The minimum possible degree of vertices is zero & maximum possible degree is n-1. Average number of edges per node in the graph specifies the average degree of a graph. The density of a graph G is the ratio of edges in G to the maximum possible number of edges in the G. Another community detection measure used widely is centrality.

It identifies the most important node or an edge within a graph. Calculate the degree of a node known as degree centrality. Depending upon the number of connected edges to a node degree centrality of a node can be measured. Average length of the shortest path between the node and all other nodes in the graph defines the closeness centrality. A node is considered to be close to all other nodes when the closest value of a node is high. Another commonly used centrality measure is betweenness centrality. There exist many shortest paths between nodes. An edge present on number of shortest paths is considered to be a connection edge between two communities. In a graph, connected components, i.e. communities can be derived by removing connecting edges. This concept is called edge betweenness. The concept of betweenness centrality can be applied to edges as well as nodes. The Girvan Newman algorithm is based on edge betweenness.

To characterize communities generated by an algorithm on the given data parameters like number of communities and modularity are used. As the number of communities increases the quality of community also increases as it provides more dense connected components and detailed information about the data. One of the important quality parameters is modularity [Girvan & Newman, 2002]. The strength of a community is measured by modularity. Modularity values can be positive or negative. A Positive value indicates presence of community. The Dense connection between the nodes indicates high modularity within the community, whereas sparse connections between nodes indicate low modularity. The general expression of modularity is

$$Q = \frac{1}{2m}\sum_{ij}(A_{ij} - P_{ij})\delta(C_i, C_j)$$

Where,
Q: Scalar valued function (ranges from -1/2 to 1)
m: the sum of the weights over all edges (in case of weighted graphs) and the total number of links (in case
of unweighted graph)
A_{ij} : Total number of edges within community
P_{ij} : Expected number of edges in community
C_i: The community to which node i is assigned.
Here, δ (C i , Cj) = 1 if $C_i = C_j$
$\delta(C_i , C_j) = 0$ otherwise

A significant difficulty with modularity approach is that it cannot detect well defined small communities when the graphs are extremely large. Details of experimental setup used for this work are as follows: All experiments were performed on core i5 @ 1.70GHz machine, 4GB RAM and 64bit operating system.

Experimental Results: Comparative performance of the three algorithms in terms of performance parameters such as number of communities, modularity and execution time is shown in Tables 2, 3 and 4 respectively for the benchmark datasets. The last rows in each of these tables show average values of the parameters. A number of communities provide insight into strongly connected nodes in the community.

Another significant parameter for evaluation of such algorithms is modularity. Community strength is measured by modularity. The community that obtains the maximal modularity is considered to be the best. Table 3 shows the modularity values generated by the algorithms for given datasets. Average of modularity with respect to the algorithm is provided in the last row.

Social network analysis needs to process large size data and thus the execution time of an algorithm is one of the important performance parameters. Table 4 shows the execution time required to identify communities by mentioned algorithms. To analyze the performance of community detection algorithm and come across the best algorithm average valuable 2, 3 and 4 are summarized in table 5.

Table 2. The performance of community detection algorithms in terms of number of communities (C)

Dataset	Algorithm		
	GN	Louvain	LPA
Enron	6	4	18
DBLP	218	217	217
Karate Club	2	4	3
Facebook	2	9	12
Average	57	58.5	**62.5**

Table 3. The performance of community detection algorithms in terms of modularity

Dataset	Algorithm		
	GN	Louvain	LPA
Enron	0.345	0.58	0.51
DBLP	0.99	0.99	0.99
Karate Club	0.359	0.41	0.32
Facebook	0.069	0.64	0.62
Average	0.441	**0.655**	0.61

From the experimentation using benchmark data sets with data of range of degree distribution it is observed that Louvain algorithm performs better than others in terms modularity. Louvain tries to optimize the modularity of network partition using greedy optimization. The modularity is first developed locally in small communities and then the smaller communities are considered as the nodes are aggregated into bigger communities iteratively until the maximum modularity is achieved. Maximum numbers of communities are detected by the LPA in the least amount of execution time. Also the modularity value generated using LPA is closer to the maximum modularity value obtained by Louvain algorithm. LPA is the simplest and time-efficient approach. Thus the overall LPA algorithm performs better than GN and Louvain.

Community analysis is carried out using different methods and parameters. The research on community analysis has different challenges as well, such as - use of community, networks with huge sizes, evaluation and visualization of communities and the need of effective identification. With the rapid growth in network sizes, detection of communities and extracting required information from them is therefore a challenging task.

Table 4. The performance of community detection algorithms in terms of execution time in seconds

Dataset	Algorithm		
	GN	Louvain	LPA
Enron	0.417	0.107	0.12
DBLP	0.05	0.131	0.022
Karate Club	0.069	0.017	0.007
Facebook	0.479	0.179	0.016
Average	0.254	0.109	**0.041**

Table 5. Average performance of community detection algorithms in terms of number of communities, modularity and runtime

Algorithm	Performance parameter		
	#C	Modularity	Runtime
GN	57	0.44	0.254
Louvain	58.5	**0.655**	0.109
LPA	**62.5**	0.61	**0.041**

To deal with these challenges good quality communities should be detected in first and foremost step in algorithms for dynamic community detection. Once communities are identified accurately, incremental data can be updated with respect to the identified communities. Among the different methods for detecting communities, the Girvan Newman algorithm is not very time efficient.

It is difficult to detect communities in large and complex networks effectively using Girvan Newman algorithm. Louvain is the most popular modularity optimization algorithm. However, its performance is not suitable for the base community detection in large dynamic social networks. Therefore, variations of Louvain algorithms like distributed [S. Ghosh et al., 2018] and parallel [Xinyu Que et al., 2015] approaches are used to deal with disadvantage of Louvain algorithm. Experimental results in this paper show that LPA performs better as compared to other algorithms for different parameters and datasets.

CONCLUSION

This paper addresses the issue of community detection in social network analysis. Community detection is one of the important aspects in static as well as dynamic data. In this paper, three well-known state-of-the-art community detection algorithms for static graphs are evaluated using four benchmark datasets. This work centered on the study of performance of algorithms for community detection with reference to different performance parameters. Experimental results bring out that Girvan Newman algorithm is not very efficient in case of large networks as compared to the other two algorithms. Louvain algorithm generates communities

with higher average modularity, but generates a slightly lesser number of communities as well as needed more execution time when compared with LPA algorithm.

The experimental results presented in this paper show that the LPA generates the maximum number of communities with modularity value closer to maximum modularity value generated by Louvain. LPA is the fastest community detection algorithm. Therefore, algorithm LPA could be useful for generating base communities in dynamic community detection algorithms for optimum processing of incremental data. In future, this research aims at using algorithms for static community detection for the efficient and accurate detection of dynamic communities.

REFERENCES

Blondel, Vincent D, Guillaume, Jean-Loup, Lambiotte, Renaud, Lefebvre, Etienne (2008) "Fast unfolding of communities in large networks". Journal of Statistical Mechanics

Cazabet, R., Rossetti, G. (2019) "Challenges in community discovery on temporal network", Temporal Network Theory, pp 181–197

Cuvelier, Etienne, Marie-Aude Aufaure (2012) "Graph mining and communities detection" Business Intelligence pp. 117-138

Despalatovic L., Vojkovi T., Vuki C (2014) "Community structure in networks: Girvan-Newman algorithm improvement" MIPRO, Opatija, Croatia, pp 997–1002

Fortunat S. (2010) "Community detection in graphs" Physics Reports, volume 486,Issues 3-5, pp.75-174

Fortunato S., Hric, D. (2016) "Community detection in networks: A user guide", Physics Reports 659, pp.1-44

Garza S. E., Schaeffer S.E. (2019) "Community detection with the Label Propagation Algorithm: A survey", Physica A: Statistical Mechanics and its Applications, Volume 534, 122058

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

Girvan M., Newman M. (2002) "Community structure in social and biological networks" , Proc. Natl. Acad. Sci. USA, pp 8271–8276

Javadi Saeed, Seyed Haji, Pedram Gharani, Shahram Khadivi (2018) "Detecting Community Structure in Dynamic Social Networks Using the Concept of Leadership", Sustainable Interdependent Networks, pp. 97-118

Lancichinetti, Andrea, Fortunato, Santo (2009) "Community detection algorithms: A comparative analysis". Physical Review E. 80: 056117

Lei Tang, Huan Liu (2010) "Graph Mining Applications to Social Network Analysis", Springer Science and Business Media, LLC

Maryam Pourkazemi, Mohammadreza Keyvanpour (2013) "A survey on community detection methods based on the nature of social networks", 3rd International Conference on Computer and Knowledge Engineering, Ferdowsi University of Mashhad

Michele Coscia, Fosca Gianno, Dino Pedreschi (2011) "A classification for community discovery methods in complex networks" Stastical Analysis and Data Mining 4, 5 pp. 512–546

Mohammad Ali Tabarzad, Ali Hamzeh (2018) "Incremental community miner for dynamic networks", Applied Intelligence, Volume 48, Issue 10, pp 3372–3393

Raghavan U.N., Albert R., Kumara S. (2007) "Near linear time algorithm to detect community structures in large-scale networks", Physical Review E 76: 036106

Xinyu Que, Fabio Checconi, Fabrizio Petrini, John A. Gunnels (2015) "Scalable community detection with the louvain algorithm", Parallel and Distributed Processing Symposium, IEEE International, pp. 28-37

Zhao Z., Li L., Zhang Z., Chiclana F., Viedma E. H.(2019) "An incremental method to detect communities in dynamic evolving social networks", Knowl.-Based Syst., vol. 163, pp. 404-415