Performance Analysis of Girvan-Newman Algorithm on Different Types of Random Graphs

Tripo Matijević, Tijana Vujičić,
Jelena Ljucović, Petar Radunović, Adis Balota
Faculty of Information Technology
University Mediterranean
Vaka Đurovića bb, 81 000 Podgorica, Montenegro
{tripo.matijevic, tijana.vujicic, jelena.ljucovic,
petar.radunovic, adis.balota} @unimediteran.net

Abstract. *A graph is an abstraction for modeling relationships between things. Different types of graph can be used to model real networks, depending on their characteristics. Main goal of this paper is to analyze performances of one of the most widely applied algorithms for clusterization of graphs, Girvan – Newman algorithm, on different types of randomly generated graphs in order to see what type of graph is the most appropriate to use in real world example.*

Keywords. Graphs, clusterization, social networks, Girvan-Newman algorithm, performance evaluation, random graphs

1 Introduction

Nowadays much information can be gained by analyzing data derived from social networks. One of growing challenges in this field is identifying of "communities" within social networks, which means identifying subsets or clusters containing the nodes (people or other entities that form the network) with unusually strong or numerous connections.

Social networks are naturally modeled as graphs, where entities are represented as nodes, and relations are represented as edges between nodes. Different type of graphs can be used to model real social network, depending on the network's characteristics.

Girvan–Newman algorithm (GN) is well known, efficient and one of the most widely applied algorithms for clusterization in social networks. Main goal of this paper is to analyze GN performances on different types of graphs (undirected, cyclic directed and acyclic directed) with different number of nodes, in order to determine which type of graph is the most suitable for clustering using this algorithm. Graphs that were used in this experiments are randomly generated, because random models are commonly used to reproduce the properties of real networks in order to analyze their behavior.

In chapter Graphs authors give brief description of graph theory and different types of graphs. Chapter Representation of social networks as graphs explains similarity between social networks and graphs, GN clustering algorithm and most important parameters of social networks that will be analyzed. Chapter Methodology contains description of used tools, technologies and algorithms. Achieved results are presented in chapter Results. In chapter Conclusion authors give their view of the analysis and propose future work on this topic.

2 Graphs

A graph is an abstraction for modeling relationships between things (Lafore, 2002). It often serves as visual and computer-friendly representation of real world data, and eases finding connections, groups, similarities etc. A graph consists of vertices or nodes, that represent real world objects, people, systems or parts of a system, and edges that connect those nodes, serving to show connection or relationship between nodes. A node can have zero or more edges, connecting it to the same number of other nodes, and that number is known as node degree.

A sequence of edges, leading from one node to another, is a path. A graph is considered connected if there is a path from any node to any other node, and thus the graph is comprised of a single connected component. There is also possibility that the graph is not connected, i.e. that it comprises of multiple connected components, lacking edges between them.

Based on the type of edges in a graph, there are two graph types:

- Undirected, where the edges of the graph do not have a direction; one can go either way on them;

- Directed, where the edges have direction determined; one can go only from node A to node B, not vice versa.

A cycle is a path that ends with the same node it began with. Directed graphs occur in two variations, in regard of cycles:

- Cyclic, that have at least one cycle in them. further in this paper they will be referred to as simply “directed”;
- Acyclic, that have no cycles whatsoever.

Graphs can represent real data, as mentioned earlier, or can be generated via certain algorithms using mathematical rules. One type of generated graphs is random graphs. A random graph consists of *N* nodes where each node pair is connected with probability *p*. (Ljucović et al., 2016)

To construct a random graph, these steps are followed:

- Start with N isolated nodes;
- Select a node pair and generate a random number between 0 and 1. If the number exceeds p, connect the selected node pair with an edge, otherwise leave them disconnected;
- Repeat second step for each of the N(N-1)/2 node pairs.

The graph obtained after this procedure is called a random graph or a random network. Two mathematicians, Pál Erdős and Alfréd Rényi, have played an important role in understanding the properties of these networks. In their honor a random network is called the Erdős-Rényi network (Erdős & Rényi, 1959).

The examples of all three types of graphs, generated randomly, are shown on fig. 1, fig. 2 and fig. 3. All graphs were generated with parameters *N*=15 and *p*=0.2.

Figure 1. Undirected graph

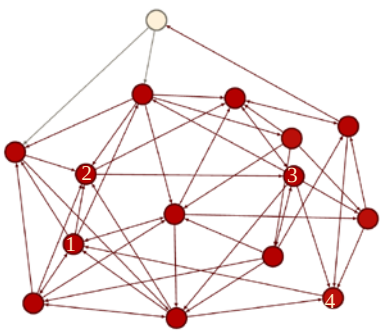

Figure 2. Directed graph, with one of the cycles marked 1-4

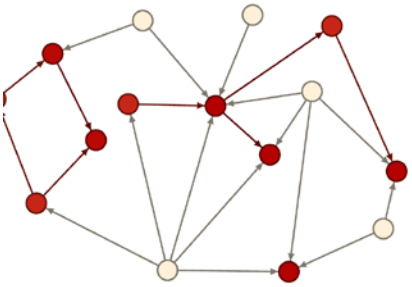

Figure 3. Directed graph, acyclic

3 Representation of social networks as graphs

From modeling aspect, social networks are simple to create – they consist of entities and relations between entities only. Thus, they can be easily modeled as graphs, where entities would be represented as nodes and relations by edges between nodes. However, the challenge is to construct a model that represents the real system, i.e. the real social network, in the best way. In that sense, philosophy of random networks is simple: the assumption is that the best representation of the real system is generated by randomly connecting any pair of nodes, so random graphs are useful in modeling social networks.

Other similarity between social networks and graphs is that social networks can be roughly divided into two types, that correspond to types of graphs mentioned earlier: directed and undirected. In directed networks existing edge from one node to another does not necessarily imply reciprocity, e.g. email networks: one email sent from an address – node to another creates a directed edge; social network twitter that has “follow” option, etc. In undirected networks existing edge marks bidirectional connection between nodes, e.g. in collaboration network where entities are authors of scientific papers, edges represent collaboration between those authors and point to existence of at least one joint paper; social network Facebook with its “friend” option, etc.

One feature that occurs prominently in social networks is clusterization. This means that there are more or less disjoint groups of nodes – clusters – that are more connected between themselves than with other nodes. In real world, this would mean that objects whose nodes fall within the same cluster are more likely to share some common features or have similarities in other way. Clusters are not necessarily recognized at the first sight, so clustering algorithms are used in order to identify them. One of the best known clustering algorithms in social networks is Girvan-Newman. (Girvan & Newman, 2002)

3.1 Girvan-Newman algorithm

The best-known algorithm for finding clusters, or in social networks terms – communities, in social networks that uses divisive hierarchical clustering is Girvan-Newman (further: GN) algorithm (Girvan & Newman, 2002). It is one of the most widely applied algorithms for social network graph clustering, based on detection of edges that are least likely to fall within the same cluster. To that purpose, the graph edge is denoted a new parameter – "betweenness". GN detects clusters by progressively removing edges from the original network. The connected components (Leskovec et al., 2007) of the remaining graph represent the resulting clusters. Connected component of a graph is a subgraph in which any two nodes are connected to each other by paths, and which is connected to no additional nodes in the supergraph.

Instead of trying to construct a measure that tells us which edges are the most central to cluster, GN focuses on edges that are most likely "between" clusters. In that purpose, GN is divided into two main phases - in the first phase it calculates betweenness for every edge in graph and in second phase it uses that betweenness to cluster the graph.

Betweenness of an edge (a, b) is the number of pairs of nodes x and y, such that the edge (a, b) lies on the shortest path between x and y. To be more precise, since there can be several shortest paths between x and y, edge (a, b) is credited with the fraction of those shortest paths that include the edge (a, b). The bigger the number, it suggests that the edge (a, b) runs between two different clusters; that is, a and b do not belong to the same cluster (Girvan & Newman, 2002).

Finally, GN clusters the graph using the calculated betweenness. It starts by removing the edges from the graph in order of decreasing betweenness: it begins with the graph and all its edges, then removes edges with the highest betweenness, as many times as it is needed, until the graph has broken into a suitable number of connected components.

3.2 Parameters of social networks

Real social networks can be described using four properties: (Maimon & Rokach, 2010)

- Node degree distribution, that shows number of edges between particular nodes;
- Growth of the big component, which is significant from aspect of determining whether the network is in critical, supercritical or connected regime;
- Clusterization coefficient, which shows relations between neighbors of a node (how much are they connected);
- Average length of shortest path between two nodes.

A. Node degree distribution

On random networks shown on fig. 1, it can be seen that some nodes have a lot of edges, while some have only a few or don't have any connected edge. These differences in node degree are caused by probability p that affects the occurrence of an edge between two nodes in a random network. Probability that a node has exactly k edges, i.e. that it has a degree of $\langle k \rangle$, for a random network with parameters (N, p) can be obtained by binomial distribution equation:

$$p_k = \binom{N-1}{k} p^k (1-p)^{N-1-k}. \quad (1)$$

Most of the real networks are sparse, which implies $\langle k \rangle \ll N$. With this condition, node degree distribution from equation (1) can be well approximated by Poisson's distribution:

$$p_k = e^{-\langle k \rangle} \frac{\langle k \rangle^k}{k!}. \quad (2)$$

Poisson's and binomial distribution describe the same distribution type and have the same properties, but are represented in different parameters. Finally, in random social model, it is expected that every individual has approximately the same number of acquaintances, which presumes exclusion of exceptions: there are no individuals that are extremely popular, having much more than average number of acquaintances, nor those that are left out of society. This leads to conclusion that in random model the degree of each node is in close proximity of $\langle k \rangle$, which does not coincide with reality. In random networks most of the nodes have similar degree and the existence of hubs, i.e. nodes with large number of edges, is excluded. On the contrary, in real networks there can be seen large discrepancy from average node degree in actual nodes, and definite existence of hubs.

B. Growth of the big component

Interesting characteristic of real social networks is existence of the big component. In real social networks a cluster that stands out from other clusters in its size can be observed. For a big component to exist as such, each node that it contains must be connected to at least another one that is member of the big component.

For small random networks p has to be large for a big component to exist, while in larger networks smaller p is required in order to cross the threshold of the big component existence, i.e. for network to reach its critical point (equation (3)).

$$p_c = \frac{1}{N-1} \approx \frac{1}{N} \quad (3)$$

C. Clusterization coefficient

Knowing the node degree does not reveal any information on connections between its neighbors, whether they are directly connected or not. Answer to this question is given by local clusterization coefficient C_i, which measures the density of edges between direct neighbors of node i (nodes that i is connected to via

single edge, also noted as nodes on distance 1 from *i*). If $C_i = 0$, then direct neighbors of node *i* do not share any edge. However, if $C_i = 1$, then every direct neighbor of node *i* is connected to all other direct neighbors. Local clusterization coefficient in random networks is given as:

$$C_i = p = \frac{\langle k \rangle}{N}. \quad (4)$$

In order to test the value of equation (4), in his paper "Network Science" Barabási showed comparative analysis of expected and real value of C_i in real social networks (Barabasi, 2016). During the research he concluded that clusterization coefficient does not shrink proportionally to 1/*N*, but still is largely dependent of *N*. Finally, he concluded that clusterization coefficient of random networks does not coincide with clustering coefficient of real networks, but real networks have much larger clusterization coefficient than expected.

D. Average shortest path length

Small world phenomenon, also known as "six degrees of separation", has significant role in network science. The phenomenon states that if one observes any two people in the world, there exists a path between them that consists of no more than six different persons, where one is acquainted to the next. In network science sense the phenomenon indicates that the path between any two nodes is short. Two questions arise from that statement: what does it mean "short" path (short in comparison to what?) and how to explain the existence of those short paths.

Both questions can be answered by observing quite simple calculation. A random network with average node degree $\langle k \rangle$ is observed. An arbitrary node in such network has: $\langle k \rangle$ neighbors on distance 1 ($d = 1$), $\langle k \rangle^2$ neighbors on $d = 2$, … $\langle k \rangle^d$ neighbors on distance *d*. More precisely, expected number of nodes on all distances up to and including *d*, from an arbitrary node is:

$$N(d) = \sum_{i=0}^{d} \langle k \rangle^i = \frac{\langle k \rangle^{d+1} - 1}{\langle k \rangle - 1}, \quad (5)$$

from which, under assumption that $\langle k \rangle \gg 1$, it can be derived that:

$$\langle d \rangle = \frac{\ln N}{\ln \langle k \rangle}. \quad (6)$$

When equation (6) is applied on real social network, it produces $\langle d \rangle = 3.28$. Therefore, it is considered that all people in the world are separated by three to four "handshakes", i.e. any person can be reached by three to four steps following relations "friend of a friend", starting with one's own friends.

4 Methodology

In this paper authors will analyze behavior and performance of Girvan-Newman on different types of graphs (undirected, cyclic directed and acyclic directed), created randomly, with different number of nodes *N* and constant probability *p*.

Dataset used for this research is created by Java application for creating random graphs, using JUNG framework (O'Madadhain, 2016). The following parameters are input into the application: number of nodes *N*, probability *p* and desired graph type. Afterwards, the application generates a random graph with given parameters.

However, each time a random network is generated, in spite of using same values for *N* and *p*, the network will look quite differently and will have at least slightly different properties (as shown on fig. 4). Considering that fact, in order to get more precise results and make them relevant for analysis, authors generated random networks for each value of *N* 15 times. In the paper authors will present the average values of tested parameters for each value of *N*.

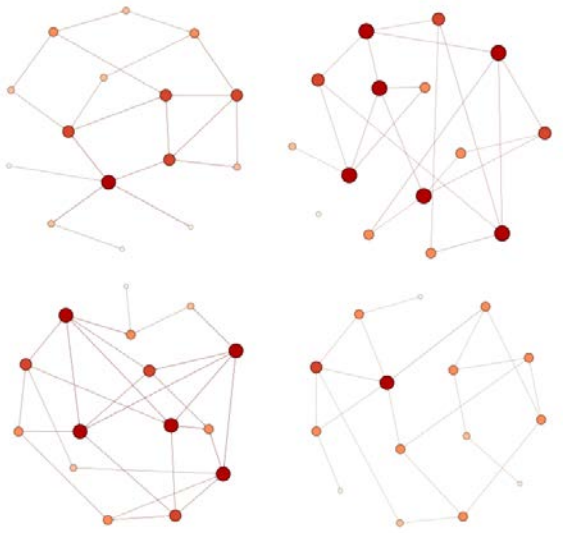

Figure 4. Randomly generated graphs with N = 15 and p = 0.2

The probability authors used in all generated graphs for this research is 0.02, because it best corresponds to probability found in real social networks. (Ljucović et al., 2016) The number of nodes that the graphs were generated with is chosen to represent possible number of inputs in the real system that this research will be used for (LAMS, 2015), and is: 500, 1000 and 1500.

In order to visualize and analyze the graphs authors used Gephi. Gephi is leading free, open source visualization and exploration software for all kinds of graphs and networks. It is written in Java on the NetBeans platform, and it is used in a number of research projects in academia, journalism, network analytics and elsewhere. (Gephi, 2016), (Kuchar, 2014)

Four main properties, described in chapter 3.2, were analyzed (node degree, clusterization coefficient, big component and shortest path length), as well as:

- Diameter, which is the longest graph distance between any two nodes in the network;
- Girvan-Newman algorithm minimal clustering level, i.e. number of components in graph;
- Number of nodes in big component, if it exists.

All tests were done on a PC with the following configuration: Windows 10 Professional, 64-bit, Intel Core i5-3317U @1.7GHz, 4 GB of RAM, Samsung EVO SSD.

5 Results

Properties of random networks are presented as average values of properties of 15 networks created for each given N. All numbers are rounded to three decimal places.

Tables 1, 2 and 3 show comparison of results that were obtained from generated networks with N=500, N=1000 and N=1500.

Table 1. Overview of results for random networks with N=500

Properties	Type of graph		
	Acyclic	Directed	Undirected
Number of edges	2516	4934	2551
Execution time of betweenness	10s	17s	13s
GN Execution time	472ms	1157ms	906ms
Number of connected components	1	1	1
Number of nodes in Big component	500	500	500
Average degree	5.032	9.868	5.102
Average clustering coefficient	0.010	0.019	0.022
Average path length	3.392	2.958	2.919
Diameter in average path length	11	5	5

Table 2. Overview of results for random networks with N=1000

Properties	Type of graph		
	Acyclic	Directed	Undirected
Number of edges	9857	19961	9894
Execution time of betweenness	17s	32s	39s
GN Execution time	1710ms	6836ms	8769ms
Number of connected components	1	1	1
Number of nodes in Big component	1000	1000	1000
Average degree	9.857	19.961	9.894
Average clustering coefficient	0.010	0.020	0.021
Average path length	3.310	2.641	2.644
Diameter in average path length	13	4	4

Table 3. Overview of results for random networks with N=1500

Properties	Type of graph		
	Acyclic	Directed	Undirected
Number of edges	22525	44793	22515
Execution time of betweenness	28s	153s	157s
GN Execution time	3280ms	23454ms	31090ms
Number of connected components	1	1	1
Number of nodes in Big component	1500	1500	1500
Average degree	15.107	29.862	15.01
Average clustering coefficient	0.010	0.020	0.020
Average path length	3.052	2.524	2.517
Diameter in average path length	11	4	4

In all three cases of N, generated graphs showed expected rise in number of edges, as it is proportional to N^2. This also had effect on average GN execution time: for N=500 it is 0.26ms per edge, for N=1000 – 0.47ms per edge, for N=1500 – 0.68ms per edge. This is noticeable increase in both absolute execution time, as can be seen in tables, but also in average, per edge. More or less all graphs were completely connected.

Average degree showed dependency on type of the graph and N, but stayed directly proportional to N. All other properties did only change slightly, and that can be attributed to statistical error. Parameters except for execution times showed similarity to those from the previous studies (Ljucović et al., 2016).

It is noticeable that GN execution time, the most changing parameter, changes with respect to N, number of edges and type of the graph. Undirected graphs have largest execution time, except in the case with fewest nodes, and show unexpectedly steeply rising curve in average execution time per edge – from 0.35ms in case of N=500 to 1.38ms in case of N=1500, while the other graph types show no noticeable change in that derived parameter.

6 Conclusion

In this paper authors give analysis of performances of Girvan-Newman clustering algorithm in different types of random graphs and show how those properties change with different number of nodes in those graphs. Also, authors present methodology and algorithms used to obtain analysis data.

Based on obtained results, it can be concluded that, expectedly, GN execution time rises with rising number of nodes, but also important observation is made regarding undirected graphs. In that case, GN execution time rose more expected, which points to possible flaw in the algorithm regarding undirected graphs with large number of nodes.

This curiosity arises a challenge in future work to find actual cause of that discrepancy, for which it will be necessary to test the algorithm on more and larger random graphs, and also, if possible, on real social networks.

For future work it is planned to use results and application mentioned in this paper as help for intelligent system that will be able to predict parameters of lightning on the mountain Lovćen for the desired time period. System described in this paper should help find similarities and group lightings.

Acknowledgments

This paper has been supported by LAMS (Lightning Activity Monitoring System) project. Results from the research behind this paper will be used in classification of lightings on Mount Lovćen station.

References

Barabasi, A. L. (2016), Network Science, Cambridge University Press, Cambridge, UK.

Erdős, P., Rényi, A. (1959), "On Random Graphs. I" (PDF). Publicationes Mathematicae

Gephi Consortium (2016), "GEPHI", available at: https://gephi.org/ (March 10, 2016).

Girvan, M., Newman, M. E. J. (2002), "Community structure in social and biological networks", in Proceedings of the National Academy of Sciences of the United States of America, pp. 7821-7826.

Kuchar, J. (2014), "The Girvan Newman Clustering plugin for Gephi", available at: https://github.com/jaroslav-kuchar/GirmanNewmanClustering (March 10, 2016).

Lafore, R. (2002), "Data Structures and Algorithms in Java", The 2nd Edition, Sams Publishing

LAMS project, http://www.lams-project.me/

Leskovec, J., Kleinberg, J., Faloutsos, C. (2007), "Graph evolution: Densication and shrinking diameters", ACM Transactions on Knowledge Discovery from Data, Vol. 1 No.1.

Ljucović, J., Matijević, T., Vujičić, T., Tomović, S. (2016) "Analysis of social network random model and comparison to real collaboration network", ITRO Conference, Serbia (unpublished)

Maimon, O., Rokach, L. (2010), Data Mining and Knowledge Discovery Handbook, Springer, USA.

O'Madadhain, J. (2016), "JUNG - Java Universal Network/Graph Framework", available at: http://jung.sourceforge.net/ (April 05, 2016).