Graph Abstraction Based on Node Betweenness Centrality

Arwa M. Aldabobi
Department of Computer Science, University of Jordan, Jordan
Email: ARO9170268@FGS.JU.EDU.JO

Riad S. Jabri
Department of Computer Science, University of Jordan, Jordan
Email: jabri@ju.edu.jo

Abstract—There are many graph abstraction methods that are existed as solutions for problems of graphs visualization. Visualization problems include edge crossings and node occlusions that hide the potential existed patterns. The aim of this research is to abstract graphs using one of network analysis metrics which is node betweenness centrality. Betweenness centrality is calculated for all graph nodes. Graph abstraction is done by removing the nodes with their attached edges such that they have betweenness centrality lower than a certain examined threshold. Experiments have been conducted and results show that the proposed abstraction method can effectively reduce the complexity of the graph visualization in term of node degree. Modularity of clusters after filtering is decreased but the final graph visualization is simpler and more informative.

Index Terms—Graph abstraction, Network analysis metrics, Nodes betweenness centrality, Degree, Modularity.

I. Introduction

Systems can be represented as a graph G (N, E), such that elements of that system are nodes N while relations between these elements are edges E [1]. An example is the Zachary's karate club in which the club members are the nodes while the friendship relations are the edges [2]. Community structure is a well-known property in systems such that community is a group of elements that has intensive relations compared to relations with another groups of elements [3]. This can give useful information about the existed patterns and the underlying structure in the system. So various community detection algorithms, also known as clustering algorithms, are existed [4].

Visualization is the process of representing graphs and clustered graphs in a pictorial view such that the system nodes, edges and detected communities are visually clear. Graph drawing problem is how to plot the graph nodes and edges in a clear and nice view, so it is easy and useful for the user to grasp information [5]. So many graph drawing algorithms are existed, and graphs are often visualized as a node-link diagram. Visualizing clustered graph should present an informative view to the user and reveal the system underlying structure by viewing distinct clusters. However, it is noticed that graphs are growing in size and being more complex in nature.

Thus representing a graph by all its elements that include important and non-important ones is not efficient in revealing the underlying structures. As a solution, abstraction, also known as filtering or simplification, algorithms are existed. Abstraction algorithms use statistical measurements in the process of filtering and highlighting the graph's important elements. The resulted graph follows the aesthetic rules, fits to be drawn using traditional layouts, clear and can be visually analyzed. There are many applications for graph abstraction, but mainly it is used to represent a clear visualization of the represented system to be easily analyzed, also to highlight and extract hidden patterns in the system.

Centrality measurements, such as betweennesss BC closeness and eigenvector, calculate the importance of a graph nodes and edges. Thus, it is a promising solution to use centrality measurements in abstracting large graphs and present a sub-graph that highlights the most effective elements in the original graph.

In this research we examine the role that nodes BC can play in abstracting graphs. First, nodes BC are calculated, second, nodes will be sorted descending in term of their calculated BC. Third, nodes that have BC below certain examined threshold will be eliminated with its attached edges from the visualization, resulting a clear graph view that reveals the underlying structure.

The paper is organized as follows: section I demonstrates the introduction, section II summarizes the most related works. The proposed method is explained in section III. Section IV presents the experimental results of the graph abstraction method and discusses the results. Finally, section V concludes the paper.

II. Related Work

Liu et al [6] present a detailed survey about summarizing graph data motivations, challenges,

summarization approaches and applications. Among the benefits, they mention the advantages of graph summarization in reducing the data volume consequently the required storage, and how the summarized graph fits the traditional graph algorithms. About the challenges that graph summarization methods face there are data volume and complexity, evaluation criteria and the dynamicity of the data. They also classify the summarization approaches into grouping or aggregation, bit compression, simplification and influence-based techniques. Graph summarization has many applications, among them they list clustering, classification and community detection.

Hu and shi [7] focus in their survey, visualizing large graphs, on issues regarding large graphs. It sheds the lights on the used algorithms in drawing large graph layout, available software and datasets, categorization of visual abstraction methods of large graphs and the challenges in visualizing large graphs. As a solution, they introduce visual representation and abstraction for visualizing large graphs on the screen.

Crnovrsanin et al in [8] try to solve hair-ball problem by using the eigenvector sensitivity analysis. Hair-ball problem is a visualization problem such as the graph edges form clusters of confusing lines. Sensitivity analysis is a centrality metric such as eigenvector, closeness and betweenness but measures the node importance in the whole network. They first abstract the graph using eigenvector sensitivity analysis, then they draw the abstracted graph using the traditional force-directed layout. Finally, they present a hierarchical abstraction of the network using modularity clustering.

Lin et al in [9] focus their study on small –world networks that are well-connected network with small diameter, short average path length and high clustering structures degree. They extract a sub-network from the original one using the network centrality metric, edges betweenness. The extracted sub-network is then drawn using force-directed algorithm. For preserving the original network integrity, they return back the filtered edges.

Zhou in [10] focus their work on human movements that are represented as origin-destination (OD) flows. They try to increase the semantic information by focusing on the intersections and occlusions between OD flows. First, they filter OD flows using edges betweenness, then edge bundling is applied. Finally, clustering using modularity is performed to extract OD flows patterns.

There are studies that also try to abstract graphs using their structural information. For example, removing one-degree nodes and duplicated paths is suggested in [11], while removing the edges with highest betweenness centrality BC from the graph to isolate and cluster its subnetwork is proposed in [12, 13]. In case of scale-free networks that do not cluster well because it have nodes that are minimally connected, removing the lowest BC edges is suggested in [14].

Despite that, abstracting graphs using the BC of its nodes has not been addressed yet. It is one of the promising abstraction methods to analyze the network in term of nodes BC that can effectively minimize the clutter in the visualized graphs.

III. Proposed Method

First, the given system is represented as a graph of nodes and edges. Each node is defined by an identification number (id) while each edge is defined by the source and target it connects them together (source and target respectively). For example, Zachary's karate club dataset is described in terms of its nodes and edges such that; nodes:{(id:0, community:0), .., (id:33, community:3)} and edges:{(source:1, target:0), (source:2, target:0), …}. Then communities within the system will be detected using fast unfolding of communities in large networks algorithm [4]. This algorithm is chosen because of its ability to detect a good quality clusters within a short computation time. In the previous example, node with (id:0) belongs to (community:0). After that, the clustered graph is drawn using force-directed algorithm [15] that plots a well-balanced graph layout and performs well with respect to minimizing edge-crossing, Fig. 2 (a).

Betweenness centrality (BC), that indicates how often a node lies on the shortest paths between other nodes pairs, is calculated for all the system nodes according to (1):

$$BC = \sum_{u \neq v \neq w \in V} \sigma_{u,w}(v) / \sigma_{u,w} \qquad (1)$$

where $\sigma_{u,w}$ counts the number of shortest paths between u and w, and $\sigma_{u,w}(v)$ counts only the ones containing v.
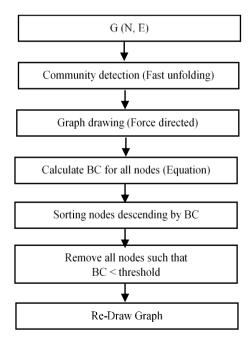

Fig.1. The proposed method's main procedures.

Then nodes are sorted descending by their BC. In the previous example this results the following, nodes:{(id:0, community:0, **BC:231.07**), (id:33, community:3, **BC:160.55**), .., (id:26, community:3, **BC:0**)}. The nodes

that have BC smaller than certain threshold are eliminated with their attached edges. About the threshold we examine two types of threshold, the first is deleting nodes that have BC equals to zero, and the second is deleting nodes that have BC less than average BC among all the graph nodes. Thus, the system visualization is clear of weak nodes and their edges, also it reveals the underlying structure. The main steps of the proposed abstraction method are summarized in Fig. 1.

IV. Results And Discussion

A. Datasets

Systems are represented as graph G (N, E) where N is the system elements and E is the relationships between these elements. An example is Zachary's karate club which is a social network of friendship between 34 members of a karate club at a US university in the 1970s. Used datasets in this research are listed in Table 1, other datasets are available in [16].

B. Results

Datasets described in Table (1) are used to study graph visualization enhancement if nodes are filtered according to their betweenness centrality. Each dataset is drawn three times; in the first time the graph is drawn using all its elements without any filtering, in the second time the graph is drawn using all its elements except nodes that have BC equals zero and their attached edges, while in the third time the graph is drawn using all its element excepts nodes that have BC less than the average BC of all nodes. Results are shown in the following tables and figures.

Table 1. Datasets description

Dataset name	Dataset description	|N|	|E|
Zachary's karate club [2]	Social network of friendships between 34 members of a karate club at a US university in the 1970s.	34	78
Coauthorships in network science [17]	Coauthorship network of scientists working on network theory and experiement.	379	914
Power grid [18]	An undirected, unweighted network representing the topology of the Western States Power Grid of the United states.	4941	6594
EuroSiSGeneralePays [19]	Mapping interactions between Science in Society actors on the Web of 12 European countries.	1272	6454

1. Zachary's karate club dataset:

When calculating BC of Zachary's karate club nodes, we find that the highest BC equals 231.07 while there are 12 nodes that have BC equals zero. This means that these 12 nodes never have been passed through when moving in the shortest path between any couple of nodes. These nodes form 35.29% of the graph nodes, Table 2. So deleting them from the visualization reveals the most important nodes that are passed through when moving from a node to another in the shortest path. Fig. 2(a) shows the graph with all the dataset elements while Fig.

2(b) shows the same dataset's graph after filtering the 12 nodes with BC equals zero. It is obvious that removing the redundant nodes clarify the visualization.

Table 2. Data related to Zachary's karate club dataset.

Number of all nodes	34
Max (BC) = 231.07, Min (BC) = 0, Avg (BC) = 23.23	
Number of nodes with BC = 0	12
(Number of nodes with BC = 0)%	35.29%
Number of nodes after filtering	22
Number of nodes with BC < avg (BC)	26
(Number of nodes with avg (BC))%	76.47%
Number of nodes after filtering	8

Although all nodes in Fig. 2(b) lies in at least one shortest path between a pair of nodes, not all of them play a controlling role in the same way. For example, the node that has BC equals 231.07 has a higher value in the graph than the node that has BC equals 0.33. So, we suggest eliminating nodes that have weak role in the graph by excluding nodes that have BC less than average BC of all nodes. Filtering using the BC average will exclude 76.47% from the nodes and the visualization shows only nodes that play the most important roles in the graph, as Fig. 2(c) shows.

It is worth to take into consideration the effects of filtering on the edges that are attached to the filtered nodes. Table 3 shows number of edges attached to the filtered nodes that have zero BC, while Table 4 shows number of edges attached to nodes that have BC less than average BC. While we mainly filter the graph from its weak and non-important nodes, also excluding their attached edges from the view present a clear graph from the crowded edges, as Fig. 2(b) and Fig. 2(c) show. In other words, it is possible to filter edges from a graph when they represent relationships between already non-important nodes in that graph.

2. Coauthorships in network science dataset:

Coauthorships dataset that has a higher number of nodes and edges acts well when filtering nodes based on their BC as Fig. 3 shows. In this figure one can notice how redundant nodes are eliminated from the final view resulting an abstracted informative graph. Table 5 lists the statistics results.

3. Power dataset:

Graph representation of power dataset has many disadvantages as Fig. 4(a) shows because the large number of drawn nodes that overlapped with each other and crossing edges that may distract the user. After calculating its nodes BC, we find that 29.28% of its nodes have a zero score, then eliminating this portion will clarify the view as Fig. 4(b) shows. Moreover, the graph has 86.72% of its nodes with BC less than the average BC of all nodes, then eliminating this portion will more
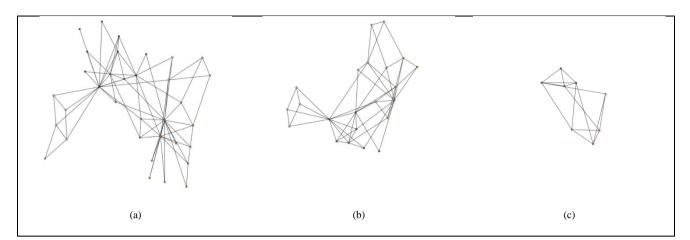
Fig.2. Karate club dataset: (a) with all its nodes, (b) after filtering nodes that have BC = 0, (c) after filtering nodes that have BC > avg (BC).
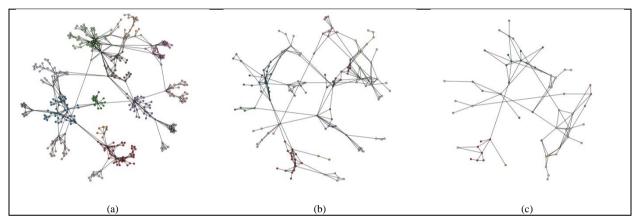

Fig.3. Coauthorships dataset (a) with a1ll its nodes, (b) after filtering nodes that have BC = 0, (c) after filtering nodes that have BC > avg (BC)
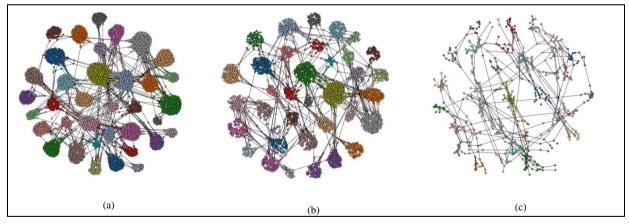

Fig.4. Power dataset (a) with all its nodes, (b) after filtering nodes that have BC = 0, (c) after filtering nodes that have BC > avg (BC).

clarify the final representation of the dataset as Fig. 4(c) shows, results are listed in Table 6.

4. *EuroSiSGeneralePays dataset:*

EuroSiSGeneralePays dataset has nature more complex than the previous datasets and this can be explained using each dataset number of nodes and edges, Table 1. Results are shown in Table (7).

For example, karate club and Coauthorships datasets have number of edges such that: $|E| \approx 2|N|$, while EuroSiSGeneralePays dataset has number of edges such

that: $|E| \approx 5|N|$. So, we can consider it as dense edge graph, and this can explain why its abstraction just based on its nodes BC cannot clarify the visualization as happen with the previous datasets. Fig. 5(b) shows the dataset after filtering out 24.29% of its nodes that have BC equals to zero while Fig. 5(c) shows the dataset after filtering out 79.48% of its nodes that have BC less than the average. Although the graph is abstracted but it still has dense edges that presents a complex graph visualization.
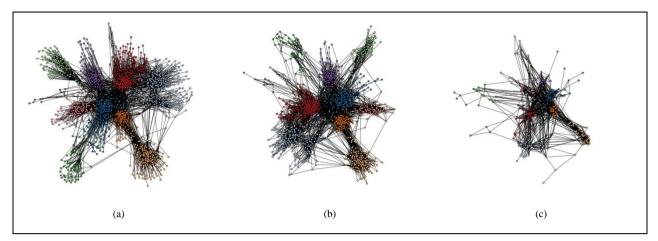
Fig.5. EuroSisGeneralePays dataset (a) with all its nodes, (b) after filtering nodes that have BC = 0, (c) after filtering nodes that have BC > avg (BC).

Table 3. Number of attached edges of nodes have (BC = 0).

Node id	Attached edges
22	2
7	4
14	2
15	2
26	2
16	2
17	2
1	**9**
12	2
20	2
21	2
18	2

Table 4. Number of attached edges of nodes have (BC < AVG).

Node id	Attached edges
19	3
5	4
6	4
27	4
23	5
30	4
3	**6**
25	3
29	4
24	3
28	3
9	2
10	3
4	3

Graphs consist mainly of its nodes and edges, in this research we focus on filtering using the graph nodes. However, in certain types of graphs such as the dense edge ones, the edges play critical role in presenting the graph. Thus, we conclude that in case of dense edge

graphs to be optimally visualized, not abstracting them based on their nodes metrics only, moreover edges metrics must be also taken in consideration. Abstracting such type of graphs can be done using both its nodes and edges metrics. We think it will be more effective in his case to combine other techniques with filtering nodes using their BC, such techniques includes for example the most used one which is edge bundling.

Table 5. Data related to coauthorships dataset.

Number of all nodes	379
Max (BC) = 28300.56, Min (BC) = 0, Avg (BC) = 952.91	
Number of nodes with BC = 0	245
(Number of nodes with BC = 0)%	64.6%
Number of nodes after filtering	134
Number of nodes with BC < avg (BC)	320
(Number of nodes with avg (BC))%	84.43%
Number of nodes after filtering	59

C. Discussion

In this section we discuss the effect of filtering process in term of changing the structure between the original graph and the abstracted one in both cases; the case in which nodes with zero BC are removed and the case in which nodes with BC less than average BC are removed.

Feature extraction is one of the proposed techniques that state that similar graphs tend to share certain properties like degree distribution, diameter and eigenvalues [20].

1. Degree:

The degree of a node is the number of edges connected to the node. In term of the adjacency matrix A, the degree for a node indexed by i in an undirected network is calculated using (2).

$$k_i = \sum_j a_{ij} \qquad (2)$$

Table 6. Data related to power dataset.

Number of all nodes	4941
Max (BC) = 3,518,477.34, Min (BC) = 0, Avg (BC) = 44,433.28	
Number of nodes with BC = 0	1447
(Number of nodes with BC = 0)%	29.28%
Number of nodes after filtering	3494
Number of nodes with BC < avg (BC)	4285
(Number of nodes with avg (BC))%	86.72%
Number of nodes after filtering	656

Table 7. Data related to EuroSiSGeneralePays dataset.

Number of all nodes	1272
Max (BC) = 93,630.36, Min (BC) = 0, Avg (BC) = 1,840.36	
Number of nodes with BC = 0	309
(Number of nodes with BC = 0)%	24.29%
Number of nodes after filtering	963
Number of nodes with BC < avg (BC)	1011
(Number of nodes with avg (BC))%	79.48%
Number of nodes after filtering	261

Average degree of the original network is compared to that in the abstracted ones when nodes with zero BC and nodes with BC less than AVG (BC), results are listed in Table 8. As it is obvious that the average degree is increased in all datasets when BC = 0 nodes are removed except in Coauthorship dataset. This could be referred to the critical role of edges in that dataset, but it is still approximately in the same dataset when BC < AVG (BC) nodes, we guess for the same reason. However, average degree value decreases in Karate dataset, we guess because the small size for this dataset.

2. Modularity:

We compute modularity that measures the quality of the graph clusters. Modularity compares edge density between nodes within a cluster with edge density with nodes outside that cluster. Modularity is computed in both cases; when filtering nodes that have (BC = 0) and when filtering nodes that have (BC < avg (BC)) using (3).

$$Q = \frac{1}{2m} \sum \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(C_i C_j) \qquad (3)$$

where A_{ij} is the weight of the edge between i and j, $k_i = \sum_j A_{ij}$ is the sum of the weight of the edges attached to vertex i, C_i is the community to which vertex i belongs, the δ – function δ(u, v) is 1 if u = v and 0 otherwise and $m = \frac{1}{2} \sum_{ij} A_{ij}$.

Because of large portion of nodes and edges are removed from the original graph, modularity is decreased in both cases, results are shown in Table 9.

3. Graph structure:

Deleting nodes that have BC < avg (BC) or BC = 0 affects negatively on modularity because there is a case in which a cluster will has only one node. So, we try to filter nodes taken into consideration maintaining the structure by excluding from filtering those nodes that will be alone in their cluster. Also, we exclud from filtering those clusters that have number of nodes less than the average number of nodes in all clusters.

Table 10 shows that excluding from filtering only single node clusters only enhance the karate club modularity but still affects negatively on modularity of the datasets comparing with their original modularity. Table 11 shows that excluding from filtering clusters that have number of nodes less than the average number of nodes per cluster for all clusters only enhances the EuroSiSGeneralPays modularity but still affects negatively on other datasets original modularity. We refer the negative effect of filtering on the resulted modularity to the modularity metric itself, since modularity takes mainly in its calculation the graph edges. On the other hand, the proposed filtering method use the graph nodes without taken its edges into consideration. Thus, filtering using the graph nodes without missing the role of its edges, especially in certain types of graphs such as the edge dense graph is a candidate solution. For example, filtering process could be implemented such that excluding important edges in the graph from filtering. Important edges could be identified using their BC values.

V. CONCLUSIONS AND FUTURE WORK

Graph visualization is considered an important tool to be used in many fields such as data mining, knowledge discovery and pattern recognition. In graph visualization, eliminating non important nodes and maintaining nodes that play a control role results an enhanced visualization of the represented system, also revealing its underlying structure. In this research, we suggest to eliminate nodes that are rarely existed in the shortest path between other graph nodes, by using nodes betweenness centrality measurement BC. Although using BC shows an abstracted view of graphs, graphs that are considered dense edge ones require another level of abstraction that take in the consideration the edges properties. As a future work, we aim to extend the proposed method by studying the enhancement on large and dense graph visualization using a hybrid abstraction method that rely on graph nodes and edges properties. Another candidate solution that can enhance filtering process is to filter nodes taken into consideration modularity value to not be decreased under certain value that could be a user-define value. We guess that this will be a promising solution for abstracting large dense graphs, in addition to preserving the graph structure that is indicated by metrics such as modularity.

Table 8. Comparison in term of degree.

Dataset	Degree (Original graph)			Degree (nodes (BC = 0) removed)			Degree (nodes (BC < AVG) removed)		
	$\sum Degree$	|nodes|	Avg (Degree)	$\sum Degree$	|nodes|	Avg (Degree)	$\sum Degree$	|nodes|	Avg (Degree)
Zachary's karate club	156	34	4.58	106	22	4.81	32	8	4
Coauthorship in network science	1828	379	4.82	562	134	4.19	188	59	3.18
EuroSiSGeneralPays	12908	1272	10.14	12022	963	12.48	3300	261	12.64
Power grid	13188	4941	2.66	9850	3494	2.81	1598	656	2.43

Table 9. Comparison in term of modularity.

Dataset	Modularity (Original graph)	Modularity (nodes (BC = 0) removed)	Modularity (nodes (BC < AVG) removed)
Zachary's karate club	0.41	0.24 (-0.17 = 41.46%)	0.17 (-0.24 = 58.53%)
Coauthorship in network science	0.84	0.39 (-0.45 = 53.57%)	0.30 (-0.54 = 64.28%)
EuroSiSGeneralPays	0.72	0.38 (-0.34 = 47.22%)	0.31 (-0.41 = 56.94%)
Power grid	0.93	0.46 (-0.47 = 48.80%)	0.41 (-0.52 = 55.91%)

Table 10. Excluding single node clusters approach.

Dataset	Modularity (Original graph)	Modularity (nodes (BC < AVG) removed)	Modularity (nodes (BC < AVG) removed Except single node in a cluster)
Zachary's karate club	0.41	0.17	0.21
Coauthorship in network science	0.84	0.30	0.28
EuroSiSGeneralPays	0.72	0.31	0.31
Power grid	0.93	0.41	0.41

Table 11. Excluding less than average node clusters approach.

Dataset	Modularity (Original graph)	Modularity (nodes (BC < AVG) removed)	Modularity (nodes (BC < AVG) removed Except less than average node clusters)
Zachary's karate club	0.41	0.17	0.17
Coauthorship in network science	0.84	0.30	0.26
EuroSiSGeneralPays	0.72	0.31	0.45
Power grid	0.93	0.41	0.25

REFERENCES

[1] William Kocay, Donald L Kreher, "Graphs, Networks and Algorithms", *Chapman and Hall/CRC*, November 2016.

[2] Wayne W. Zachary, "An Information Flow Model for Conflict and Fission in Small Groups", *Journal of Anthropological Research*, Vol. 33, No. 4, pp. 452-473, December 1977.

[3] Mark Newman, "Modularity and Community Structure in Networks", *Proceedings of the national academy of sciences*, Vol. 103, No. 23, pp. 8577-8582, June 2006.

[4] Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte, Etienne Lefebvre, "Fast Unfolding of Communities in Large Networks", *Journal of statistical mechanics: theory and experiment*, July 2008.

[5] Ivan Herman, Guy Melancon, M. Scott Marshall, "Graph Visualization and Navigation in Information Visualization: a Survey", *IEEE Transactions on visualization and computer graphics*, Vol 6, No. 1, pp. 24-43, January 2000.

[6] Liu Y., Safavi T., Dighe A. and Koutra D., "Graph Summarization Methods and Applications: A Survey", *ACM Computing Surveys (CSUR)*, Vol. 51, No. 3, July 2018.

[7] Hu Y. and Shi L., "Visualizing Large Graphs", *Wiley Interdisciplinary Reviews: Computational Statistics*, Vol. 7, No. 2, pp. 115-136, 2015.

[8] Crnovrsanin T., Muelder C.W., Faris R., Felmlee D. and Ma K.L., "Visualization Techniques for Categorical Analysis of Social Networks with Multiple Edge Sets", *Social Networks*, Vol. 37, pp. 56-64, 2014.

[9] Lin C.C., Huang W., Liu W.Y. and Wu S.F., "A Novel Centrality-Based Method for Visual Analytics of Small-World Networks", *Journal of Visualization*, pp. 1-18, 2019.

[10] Zhou Z., Meng L., Tang C., Zhao Y., Guo Z., Hu M. and Chen W., "Visual Abstraction of Large Scale Geospatial Origin-Destination Movement Data", *IEEE transactions on visualization and computer graphics*, Vol. 25, No. 1, pp. 43-53, 2018.

[11] Zeqian Shen, Kwan-liu Ma, Tina Eliassi-Rad, "Visual Analysis of Large Heterogeneous Social Networks by Semantic and Structural Abstraction", *IEEE transactions on visualization and computer graphics*, Vol. 12, No. 6, pp. 1427-1439, September 2006.

[12] Michelle Girvan, Mark Newman, "Community Structure in Social and Biological Networks", *Proceedings of the national academy of sciences*, Vol. 99, No. 12, pp. 7821-7826, June 2002.

[13] M. E. J. Newman, "Detecting Community Structure in Networks", *The European Physical Journal B-Condensed Matter and Complex Systems*, Vol. 38, No. 2, pp. 321-330, June 2004.

[14] Yuntao Jia, Jared Hoberock, Michael Garland, John C. Hart, "On the Visualization of Social and Other Scale-Free Networks", *IEEE transactions on visualization and computer graphics*, Vol. 14, No. 6, pp. 1285-1292, 2008.

[15] Peter Eades, "A Heuristic for Graph Drawing", *Congressus numerantium*, Vol. 42, pp. 149-160, 1984.

[16] http://www-personal.umich.edu/~mejn/netdata/, accessed in January 2019.

[17] M. E. J. Newman, "Finding Community Structure in Networks using the Eigenvectors of Matrices", *Physical Review E*, Vol. 74, No. 3, September 2006.

[18] Duncan J. Watts, Steven H. Strogatz, "Collective Dynamics of 'Small-World' Networks", *Nature*, Vol. 393, pp. 440-442, June 1998.

[19] https://github.com/gephi/gephi/wiki/Datasets, accessed in January 2019.

[20] Danai Koutra, Ankur Parikh, Aaditya Ramdas, "Algorithms for Graph Similarity and Subgraph Matching", *Proc. Ecol. Inference Conf*, Vol. 17, 2011.

Authors' Profiles

Arwa M. Aldabobi PhD candidate in computer science at University of Jordan (Amman/Jordan). She received her first degree in computer engineering from Al Balqa Applied University, Jordan, 2010, master's degree in computer science from University of Jordan, Jordan, 2017. Her research focuses on pattern recognition and information visualization.

Riad S. Jabri Professor in Programming languages and compiler design at University of Jordan (Amman/Jordan).