Scalable Community Detection through Content and Link Analysis in Social Networks

Zahra Arefian*
Department of Computer Engineering, University of Isfahan, Isfahan, Iran
zahra_arefian@yahoo.com
Mohammad Reza Khayyam Bashi
Department of Computer Engineering, University of Isfahan, Isfahan, Iran
m.r.khayyambashi@eng.ui.ac.ir

Received: 16/May/2015 Revised: 29/Sep/2015 Accepted: 28/Oct/2015

Abstract

Social network analysis is an important problem that has been attracting a great deal of attention in recent years. Such networks provide users many different applications and features; as a result, they have been mentioned as the most important event of recent decades. Using features that are available in the social networks, first discovering a complete and comprehensive communication should be done. Many methods have been proposed to explore the community, which are community detections through link analysis and nodes content. Most of the research exploring the social communication network only focuses on the one method, while attention to only one of the methods would be a confusion and incomplete exploration. Community detections is generally associated with graph clustering, most clustering methods rely on analyzing links, and no attention to regarding the content that improves the clustering quality. In this paper, a novel algorithm for community selection is proposed. Scalable community detections, an integral algorithm is proposed to cluster graphs according to link structure and nodes content, and it aims finding clusters in the groups with similar features. To implement the Integral Algorithm, first a graph is weighted by the algorithm according to the node content, and then network graph is analyzed using Markov Clustering Algorithm, in other word, strong relationships are distinguished from weak ones. Markov Clustering Algorithm is proposed as a Multi-Level one to be scalable. Finally, we validate this approach through a variety of data sets, and the effectiveness of the proposed method is evaluated.

Keywords: Social Networks; Community Detections; Link Analysis; Clustering; Scalable.

1. Introduction

In recent years, social networks have not only been being used for creating relationships, but they are also used to share opinions, communicate, fans, activists and interact over diverse geographical regions [1]. Due to the multiple modes of communication, these networks share information and do a variety of interactions. These relationships will lead to the creation of groups like friends, colleagues, acquaintances, family and other similar groups, and that's why social networks have been popular. According to a report in 2012, internet users spent 22 percent of their online time surfing social networks.

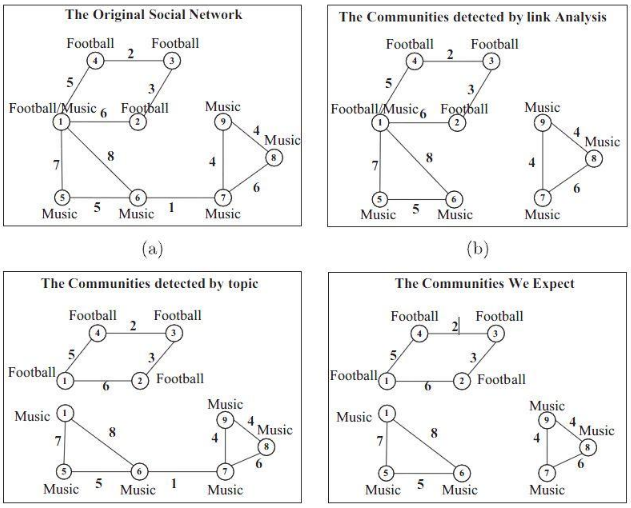
Among the popular social networks, we can mention Facebook[1], YouTube[2], Flickr[3], twitter[4], etc [2]. Social networks are a social structure; a social network is a network of interactions and relationships that are a graph and set of nodes and edges (nodes consisting of individuals or organizations). These nodes have interactions and according to the social relations that exist in the real world, these relations can be obtained and we can analyze them (links represent the connections between users) [3]. Due to the amount information of data in these networks, the analysis of network data has become an important issue for research. Now, according to the large volume of information, we should discover the unknown relations, and the discovery can be exploited to improve opportunities. Despite the increasing significance and complexity of Social network, there has expanded of methods for detecting communities. The discovery of communication by link analysis and regarding the content of nodes is an important issue.

The importance of addressing the link analysis and the nodes content for community detection is illustrated in Fig. 1 [1]. In Fig. 1(a) presents a very small social network. The nodes indicate the number of involved members in the social activities and the edges represent the social relations and interactions among members. The weight wrote to each edge illustrates the strength of connections between the corresponding members and also each node is labeled according to its interests. Fig. 1(b) presents the result of discovered communities based on link analysis, that the discovery relates to the link analysis, they only pay attention to the network topological structure or analysis from respect data mining [4]. Fig. 1(c) presents the result of discovered communities based on nodes content, they only pay attention to the "Similarity theory" for categorizing individuals with different

[1] www.facebook.com
[2] www.youtube.com
[3] www.flickr.com
[4] www.twitter.com

* Corresponding Author

communication; but this type of grouping doesn’t have enough accuracy and can not create a strong social relations. Fig. 1(d) presents the desired grouping, the groups are defined to having the same interest topics and strong relation between each cluster.


The Original Social Network

The Communities detected by link Analysis

The Communities detected by topic

The Communities We Expect

Fig. 1. An example that in part(a) presents a very small social network. Part(b) presents the result of discovered communities based on link analysis. Part(c) presents the result of discovered communities based on nodes content. Part(d) presents community detection based on node content and link analysis[1].

large social networks challenge the issue of discovering the communications, and the old methods of discovering such communications have a lot of problems.

By modeling complex social network with graphs, the community detection can be modeled by graph partitioning, and there is a clustered set of nodes which are connected by the edges. Although the number of different clustering algorithms exploring the relationship exists, but it is not easy to present a good algorithm for the above cases, and it requires careful consideration.

In this paper, an algorithm for clustering the graph topology structure is presented according to the vertices features, and a graph is formed according to the features and content which are among users, then the scalable communications will be discovered using Multi-Level Markov clustering (ML-MCL) algorithm.

The article is organized as follows; first, in part 2, reviews the related work, then will be propound the subject and the proposed approach in the part 3, finally, Performance assessment methodology and conclusions will have been done.

2. Related Work

Social networks has been a very important matter in recent decades, so a lot of fundamental and important research has been done in all fields and topics; that is because these networks are posing global communications. In Ref [5] One of the most important topic that researchers has been working on it is exploring the community in the social networks, that some of them are mentioned in the following expression.

In Ref [6] Markov Clustering (MCL) Algorithm groups nodes randomly, and clusters graphs via transition probability matrix corresponding to the graph. The MCL algorithm is an iterative process of applying two operators (expansion and inflation) in alternation, until convergence. Additionally, a prune step is performed at the end of each inflation step in order to save memory. One of the important algorithm used for community detection is KL[1] algorithm that is graph partitioning algorithm and is run in classic way and do optimization operations. In Ref [7] Another group of Algorithms for community detection are Agglomerative/Divisive Algorithms. Agglomerative algorithms at first begin with each node in its own community, and at each step communities merge each other, continuing till either the desired number of communities is obtained or the remaining communities don’t have enough similarity for merging. Divisive algorithms operate in reverse. Both types of algorithms are hierarchical clustering algorithms and their output is a type of binary tree. The other way which is mainly used to for community detections is the Local Graph Clustering which is used to reduce the scalability challenges by focusing on the studying section of the network. To discover, it is started from a peak as a seed and then by adding the neighbor peaks to the community, it is resulted to increase the network and obtain a high-quality proper size in [8-10]. Another groups of Algorithms for community detection are Spectral algorithms. Generally, assign nodes to communities based on the eigenvectors of matrices, such as the adjacency matrix or other related matrices. Spectral methods aim to minimize the defined cut-function that lead to more resolution in graph clustering structure in [11-12]. Multi-level algorithms are of the other algorithms which are used to discover the communications. Multi-level methods present a framework for high-quality, fast partitioning of a graph and are used to solve many problems. The main idea is to minimize the input graph continuously and reach a smaller graph. The resulted graph is partitioned and then returned to the top to reach the main graph. Some methods to partition multi-level graphs are multi-level spectral clustering, Metis (improved KL function) and Graclus (improved normal cut and weight loss) [13-15]. In [16], at first, they develop the original similarity based on the social balance theory. Then, based on the natural contradiction between positive and negative links and the signed similarity, two functions are designed to model a multi objective problem, called MEAs -SN. In [17], Based on the Max-Flow Min-Cut theorem, they propone a novel algorithm which can output an optimal set of local communities automatically.

[1] Kernighan-Lin Algorithms

3. Community Detection Mechanism

In this section, the proposed mechanism of Community Detections is presented; the work done in this section is as follows:

First, graph topology structure is combined with node features, then edges are weighted according to vertices content, and links are analyzed by MCL algorithm according to the weighted graph; On the other hand, MCL clustering algorithm is proposed as a multi-level one to be scalable.

3.1 Social Network Data Modeling Based on Similarities

Social networks are shown in graph G= (V,E,χ), that $V=\{v_1, v_2, ..., v_n\}$ is the set of nodes and $|V| = n$ illustrate the number of persons in graph. Also $E \subseteq V \times V$ is the set of edges, where $E = \{(v_i, v_j): v_i, v_j \in V\}$ and shows collection of interactions and communications among individuals. In this graph, matrix χ is attributes of vertices and $\chi \in R^{|V|\times d}$, where d indicates number of node attributes [4].

Similarity function C determines the similarity between each pair of vertices in an attributed graph G. all of the characteristics are binary, so we use Jaccard's coefficient as similarity criteria for attributes data that is eq. 1:

$$J(V_i, V_j) = \frac{\text{Number of common one's between i and j}}{\text{Number of attributes}} \qquad (1)$$

Then based on vertices content matrix S is constituted, so if content of vertices v_i and v_j are similar, according to the number of topics that are interacted to each other S_{ij} from matrix S are calculated power of link and if they do not have any interaction with each other is placed 0. Finally weight matrix W is collection of matrix S and matrix A, that is eq. 2:

$$W=A+S \qquad (2)$$

Until this step, social network graph is weighted based on vertices content. In the following we describe the proposed clustering Algorithm for grouping social topics.

3.2 The Clustering Algorithm

In this section, clustering algorithm to Community Detections in social networks through vertices content and link analysis has been proposed; following steps is required in this algorithm:

Pseudo-code of the integral clustering algorithm to discover communications in social network graphs is presented in Fig. 2. In this pseudo-code, the social network graph is used as the algorithm input and after a few steps; a clustered graph is returned as an output.

```
1. Input: G
2. Output: clustering
3. A ← Adj(G)
4. Compute the attributes similarity matrix, C
5. Compute matrix S
6. W ← A + S
7. clusters ← Apply Multy-Level Markov Clustering on W
8. Return clusters
```

Fig. 2. Clustering Algorithm based on MCL[4]

First, a similarity matrix C is developed, then matrix S is formed in the fifth line of the algorithm, next matrix W is formed by adding the matrix S and matrix A. Finally, in the seventh line of the algorithm, the formed weighted graph using the multi-level clustering algorithm of Markov (ML-MCL) is clustered. In the following, the pseudo-code related to each one is presented.

Fig.3 shows the pseudo-code of MCL algorithm, and Fig. 5 shows ML-MCL algorithm which is obtained by some changes in original pseudo-code of MCL algorithm.

```
1. A := A + I // Add self-loops to the graph
2. M := AD−1 // Initialize M as the canonical
   transition matrix
3. repeat
   a. M := Mexp := Expand(M)
   b. M := Minf := Inflate(M, r)
   c. M := Prune(M)
4. until M converges
5. Interpret M as a clustering
```

Fig. 3. MCL Algorithm[18]

MCL algorithm is a clustering algorithm based on graph stochastic flows simulation. The reasons to choose this algorithm for some of the clustering steps are that this algorithm has no need to specify the number of clusters at the beginning, and has resistance to noise in the number of components, also its efficacy for weighted and non-weighted graphs and oriented and non-oriented ones. First, to conclude more quickly and prevent some unexpected cases, the first line of the algorithm is done for convergence. Then, to implement it, the transition matrix should be formed from the weighted graph W [18]. To calculate the components of the transition matrix, eq. 3 is implemented:

$$M_{ij} = \frac{A_{ij}}{\sum_{i=0}^{n} (A_{ij})} \qquad (3)$$

The resulted transition matrix is a type of the column-stochastic transition matrix, a matrix that the sum of each of its columns equals to 1. Such matrices can be defined as transition matrix of Markov chain in which the i^{th} column of matrix M represents the possibility of transferring the output V_i. Therefore, M_{ij} represents the possibility of transferring V_i to V_j. In the transition matrix M, the i^{th} column includes the flow of the output V_i and the i^{th} row includes the input flow to V_i. So, the sum of the elements of each column equals to 1, but this rule is not true for every row [19].

The process of the algorithm MCL includes two expand and inflate operators on the random matrix; and this is continued until the matrix is converged. In addition, there is also a prune step in end of each inflate step to save memory and increase speed, which are addressed as bellow:

Expand calculates the square of the matrix M as $M_{exp} = M * M$, which is a factor to transfer power according to Markov chain and lets the different regions in a graph to be connected.

Inflate increases each element of the matrix M to the value of the inflate parameter r (r>1), and then normalizes the columns of the matrix so that the sum of the existing entries in each columns is 1. How inflate is calculated for each matrix element is presented in eq. 4:

$$M_{\text{inf}}\,(i,\, j) = \frac{M(i,j)^r}{\sum_{k=1}^{n} M(k,j)^r} \qquad (4)$$

The parameter r is considered as 2 which causes strong flows become stronger, and weak flows become weaker. Therefore, the Inflation equation will become eq. 5:

$$M_{\text{inf}}\,(i,\, j) = \frac{M(i,j)^2}{\sum_{k=1}^{n} M(k,j)^2} \qquad (5)$$

The last step of the MCL algorithm is prune so that very small values are emitted to reduce the used memory and calculation operation. To do so, a threshold in considered and the values smaller than it are considered as zero [18].

In the MCL algorithm, with the beginning of a standard flow matrix, then algorithm is an iterative process of applying two operators - expansion and inflation - on a matrix, until the output matrix reaches the steady state M^{∞} and after that applying these two operators has no effect on the output matrix.

Up to this step, the given idea, community detections based to content and link analysis, is done. By studying the MCL algorithm, it has certain features to the spectral clustering algorithm and heuristic clustering algorithm, but the algorithm speed has no proper scalability for the large networks, on the other hand, graphs have lower speed in early iterations of the MCL algorithm due to fewer zero values, and if it is done on a smaller graph, algorithm speed is considerably increased. So, in the following, the algorithm is presented in a scalable manner with some changes. So, ML-MCL algorithm is proposed. The general design of a multi-level algorithm and its illustration are presented in the next section.

3.3 Multi-Level Markov Algorithm to Scalab

At first, the general framework of the multi-level algorithm is shown in Fig. 4 for better understanding.

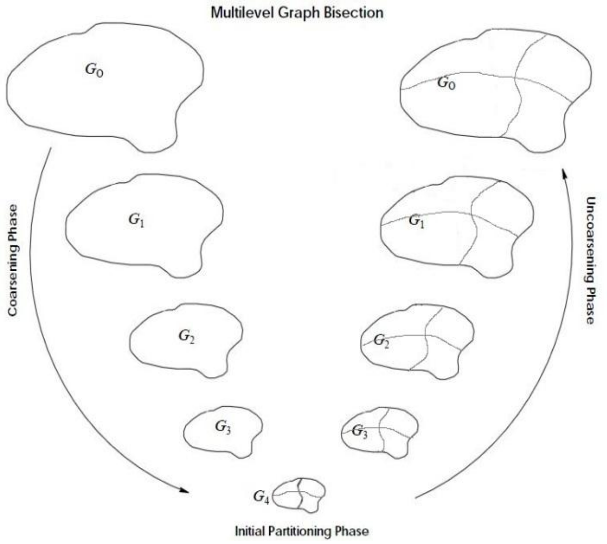


Fig. 4. doing a multi-level algorithm has three main steps; Coarsening, primary grouping, Uncoarsening. In this figure, algorithm has 4 levels, and it is divided into 4 groups in the primary grouping of the graph, then the main graph is resulted by implementing uncoarsening level[20].

As it is observed in Fig. 5, the algorithm is implemented in three levels: coarsening, primary grouping and uncoarsening, which each one is briefly described in the following:

1. Coarsening: the input of the level is the main graph G (a graph which has been weighted in the previous steps). And it is frequently divided into smaller graphs $G_1, G_2, G_3, \dots G_1$, in which $|V_0|>|V_1|>|V_2|>\dots>|V_1|$. This minimizing is continued until G_1 size can be controlled. In each level of this step, graph nodes are merged with each other and formed a super-node and sent to the next level. The ways to go from G_1 to G_1 are different of which different types are described in [13]. Coarsening utilizes the maximum matching to maintain the main graph properties. The time required to calculate these levels is O(log(n'/n)) in which n is the number of the peaks in graph G_0 and n' is the number of the peaks in G_1. In each level of coarsening, three steps are done to convert graph G_i to G_{i+1}: the first step is considered a subset of nodes to convert to a super-node according to coarsening method (this choice can be done according to the strengths and weaknesses of the interactions, randomly or with other factors). In this paper, the criterion, the most similarity is considered to merge the nodes. In the second step, the rules required to merge are applied, and in the third one, the edge weights are calculated according to the new nodes [20].
2. Primary grouping: in this step, the MCL algorithm is iterated on G_1 with few times (e.g. 4 or 5 iterations) by starting from the graph $G_1=(V_1;E_1)$ of the previous step. The reason to implement the algorithm for few times in this step is only controlling the graph distribution, and in this step, obtaining a balance is not considered. How the MCL algorithm works is fully described in the previous section. There is no problem according to the fact that the MCL algorithm did not have a proper scalability but because the graph size is not minimized in this step and on the other hand the algorithm work properly in small-sized graphs.

Uncoarsening: in the step, the multi-level algorithm is the goal to obtain the main graph by decomposing super-nodes and forming its primary node while the grouping done in the previous step is maintained. Finally, when the main graph is formed, the MCL algorithm is implemented to obtained convergence.

4. Experiment

In the previous section, the proposed strategy was presented to community detections using node content and link-analysis in social networks. In this section, Facebook real world datasets[1] was used to evaluate the efficacy of the proposed method, in which the number of the nodes is 4039

[1] http://snap.stanford.edu/

and the number of the edges is 88234. Generally, the datasets content is divided into three educational, work, location and sections; and each of these sections presents the trend of users to different groups, which represents difference in interests and properties of users. Consider the following four scenarios; the first scenario corresponds to the educational Facebook social network dataset, the second scenario is the location, third scenario corresponds the fields of work Facebook social network dataset and the fourth scenario considers all fields. Evaluations were done in a dual-core system with a 4GB main memory and processing speed 2.53 GHz.

```
Input: Original graph G, Inflation parameter r, Size of coarsest graph c
// Phase 1: Coarsening: Coarsen graph successively down to at most c nodes.
{$G_0$, $G_1$, . . . , $G_k$} = CoarsenGraph(G, c)
// $G_0$ is the original graph and $G_k$ is the coarsest graph
// Phase 2: Curtailed MCL along with refinement
// Starting with the coarsest graph, iterate through successively refined graphs.
// Run MCL for a small number of iterations.
for small number of iterations do
        Markov Clustering
end for
// Phase 3: UnCoarsening graph successively access to original graph.
Run MCL on original graph until convergence
repeat
        Markov Clustering
until M converges
```

Fig. 5. Details of ML-MCL Algorithm

4.1 Experiment Criterions

The sachan's models algorithm and the MCL algorithm were selected because of the similarity the algorithms are presented.

The first criterion to study the quality of clusters is similarity measurements that has been scaled by entropy. Well, low entropy means the high similarity between clusters and homogeneous clusters, and high entropy means there is no similarity. After checking the integral algorithm, sachan's models algorithm and the MCL algorithm, According to this criterion the results will be shown by Fig. 6, as you see, the proposed algorithm has low entropy. One of the reasons that the MCL algorithm entropy is higher than the integral algorithm entropy is only paying attention to the network structure for clustering.

Another criteria is the number of clusters that has been assessed. If the number of the cluster are much more, it causes Fragmentation in network graph in clustering, and it causes low communication discovering and high clustering. There is no paying attention to this subject in the MCL algorithm while in the integral algorithm we have done some reforms and as a result we find high coherence and thematic similarities. Evaluation results are shown in Fig. 7.

Another criterion is normalized cut or conductance that has been for evaluating of cluster quality. The normalized cut of a cluster is simply the number of edges that are "cut" when dividing this cluster from other clusters. The Normalized Cut criterion has been the quality of clusters . The normalized cut of a cluster C in the graph G is defined as eq. 6. The average normalized cut of a clustering is the average of the normalized cuts of each of the constituent clusters.

$$\text{N cut (C)} = \frac{\sum_{v_i \in C, v_j \notin C} A(i,j)}{\sum_{v_i \in C} \text{degree}(v_i)} \qquad (6)$$

Evaluation results are shown in Fig. 8, that the Integrative algorithm is presented better than MCL algorithm. Generally, the sachan's models algorithm is very close to our approach, but according to the figures proposed method is clearly effectiveness.

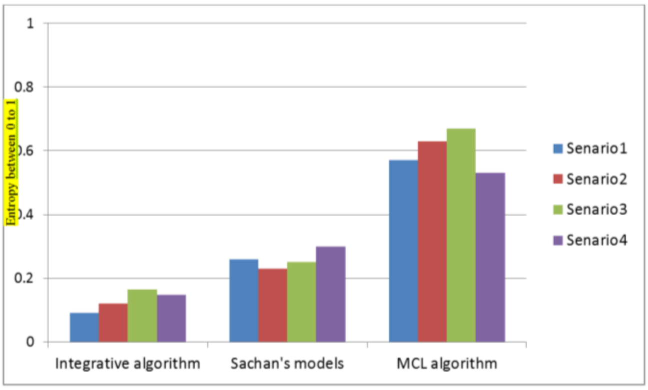


Fig. 6. Entropy

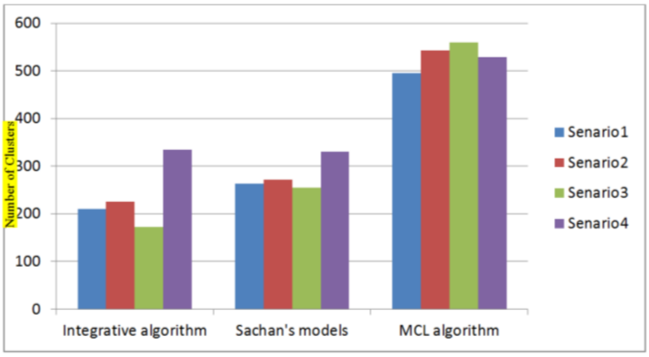


Fig. 7. Number of Clusters

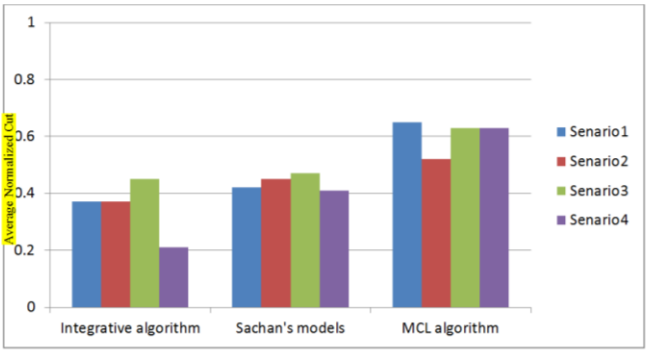


Fig. 8. Average Normalized Cut

5. Conclusions

The first obstacle to threaten many clustering algorithm is large graphs. Many clustering algorithms, including restrictions like directional network are not considered. But in the clustering algorithms, for simplicity, the direction of network graphs is not considered. In addition, combining the link and content analysis method in the same time to get a better clustering has been noted less. But all of these issues have been considered in integrative algorithm.

According to the evaluation, The clusters are homogeneous and dense , so it is clear that the integral algorithm is better than MCL algorithms and sachan's models algorithm, and it can be used to explore the communication between the social networks (weighted or non- weighted and directional or non-directional). Developing this algorithm and attention to overlapping nodes are future work.

References

[1] M. Sachan, D. Contractor, T. A. Faruquie, and L. V. Subramaniam, Using content and interactions for discovering communities in social networks, in Proceedings of the 21st international conference on World Wide Web, 2012, pp. 331–340.

[2] Z. Zhao, S. Feng, Q. Wang, J. Z. Huang, G. J. Williams, and J. Fan, Topic oriented community detection through social objects and link analysis in social networks, Knowledge-Based Systems, Vol. 26, pp. 164–173, 2012.

[3] S. Wasserman, Social network analysis: Methods and applications, Vol. 8. Cambridge university press, 1994, pp. 1-27.

[4] S. Salem, S. Banitaan, I. Aljarah, J. E. Brewer, and R. Alroobi, Discovering Communities in Social Networks Using Topology and Attributes, in Machine Learning and Applications and Workshops (ICMLA), 2011 10th International Conference on, 2011, Vol. 1, pp. 40–43.

[5] S. Dongen, A new cluster algorithm for graphs, Center for Mathematics and Computer Science (CWI), Amsterdam, 1998.

[6] B. W. Kernighan and S. Lin, An efficient heuristic procedure for partitioning graphs, Bell system technical journal, Vol. 49, No. 2, pp. 291–307, 1970.

[7] M. E. Newman and M. Girvan, Finding and evaluating community structure in networks, Physical review E, Vol. 69, No. 2, p. 026113, 2004.

[8] S. Papadopoulos, Y. Kompatsiaris, A. Vakali, and P. Spyridonos, Community detection in social media, Data Mining and Knowledge Discovery, Vol. 24, No. 3, pp. 515–554, 2012.

[9] A. Clauset, Finding local community structure in networks, Physical review E, Vol. 72, No. 2, p. 026132, 2005.

[10] F. Luo, J. Z. Wang, and E. Promislow, Exploring local community structures in large networks, Web Intelligence and Agent Systems, Vol. 6, No. 4, pp. 387–400, 2008.

[11] M. Fiedler, Algebraic connectivity of graphs, Czechoslovak Mathematical Journal, Vol. 23, No. 2, pp. 298–305, 1973.

[12] S. Smyth and P. Smyth, A spectral clustering approach to finding communities in graphs, in SDM, Vol. 5, 2005, pp. 76–84.

[13] G. Karypis and V. Kumar, A fast and high quality multilevel scheme for partitioning irregular graphs, SIAM Journal on scientific Computing, Vol. 20, No. 1, pp. 359–392, 1998.

[14] I. S. Dhillon, Y. Guan, and B. Kulis, Weighted Graph Cuts without Eigenvectors: A Multilevel Approach, IEEE Trans. Pattern Anal. Mach. Intell, Vol. 29, pp.1944–1957, 2007.

[15] S. T. Barnard and H. D. Simon, Fast multilevel implementation of recursive spectral bisection for partitioning unstructured problems. Concurrency: Practice and Experience, Vol. 6, No. 2, pp. 101–117, 1994.

[16] Liu, Chenlong, J. Liu, and Zh. Jiang, A multiobjective evolutionary algorithm based on similarity for community detection from signed social networks, *Cybernetics, IEEE Transactions* vol. 44, pp. 2274-2287, 2014.

[17] Qi, Xingqin, et al, Optimal local community detection in social networks based on density drop of subgraphs, *Pattern Recognition Letters* 36, pp. 46-53, 2014.

[18] S. M. Van Dongen, Graph clustering by flow simulation, PhD Thesis, University of Utrecht, 2000.

[19] V. M. Satuluri, Scalable Clustering of Modern Networks, The Ohio State University, 2012.

[20] C. Chevalier and I. Safro, Comparison of coarsening schemes for multilevel graph partitioning, in Learning and Intelligent Optimization, Springer, 2009, pp. 191–205.

Zahra Arefian received the M.Sc. degree in Computer Architecture Engineering from Isfahan University, Esfahan, Iran, in 2014. She received the B.Sc. degree in Computer Hardware Engineering from HaMedan University of Technology, Hamedan, Iran, in 2012. Her area research interests include Social Network.

Mohammad Reza Khayyam Bashi received Ph.D. degree in Department of Computing Science, University of Newcastle Upon Tyne, Newcastle Upon Tyne , England in 2006. He received the M.Sc. degree in Computer Architecture Engineering from Sharif University of Technology, Tehran, IRAN in 1990. He received the B.Sc. degree in Computer Hardware Engineering from Tehran University, Tehran, IRAN in 1987. His area research interests include Distributed Systems, Computer Networks, Fault Tolerance.