Article

An influence maximization algorithm in social network using K-shell decomposition and community detection

Alighanbari, **Esmaeil Bagheri**
Department of Computer, Dehaghan Branch, Islamic Azad University, Isfahan, Iran
E-mail: Bagheri471@gmail.com

Received 11 November 2019; *Accepted 15 December 2019*; *Published 1 March 2020*

Abstract
The increasing use of services and different applications of social networks has led to a wide range of research and studies in the field of information technology and computer networks towards such networks. Creating a wide platform for advertising in social networks and attracting more customers in this way has created a variety of ways to maximize profits. Therefore, due to the high importance of the propagation speed and the extent of advertising, the issue of influence maximization is considered special. The influence maximization can be described as: determining a small set of nodes capable of operating large waterfalls of behavior that are spread across the network. In other words, selection of a set of K nodes from a social network is in such a way that the influence of the node in the network has maximum value. Due to the high sensitivity of the influence maximization process, in this study we try to reduce the strengths and problems of previous strategies in this field by K-shell decomposition and community detection based on SLPA algorithm. The proposed approach in this research is based on the recognition of community based on SLPA algorithm, to make a better result by flexible and optimizing the decision making in exploration and extraction of societies. In both methods, K-shell analysis and community detection are used to choose the more influential nodes, which are proportional to the graph of social networks. The proposed method is evaluated based on two fundamental criteria of execution time and number of active nodes, which have better efficiency and efficiency compared to previous methods.

Keywords social networks; community detection; spread of influence; influence maximization.

Network Biology
ISSN 2220-8879
URL: http://www.iaees.org/publications/journals/nb/online-version.asp
RSS: http://www.iaees.org/publications/journals/nb/rss.xml
E-mail: networkbiology@iaees.org
Editor-in-Chief: WenJun Zhang
Publisher: International Academy of Ecology and Environmental Sciences

1 Introduction

Social networks are comprised of individual nodes that exchange information through cooperation, conversation and friendship (Zhang, 2018). These networks play an important role in information broadcast. Damingos and Richardson offered influence maximization in social networks to improve marketing efficiency (Domingos and Richardson, 2001). This problem helps finding set of k nodes which are the most influential nodes based on spread model and it is an important problem associated to improving quality of products. For instance, in order to sell a new product with limited price, a company can select a limited number of customers in a social network and offer them free instances of the product and expect them to introduce the product to their friends and influence behavior of their friends and then their friends also influence their friends; thus, their product is advertised widely. In this method, many of the nodes (customers) adopt a new product for advertisement. Many of the customers are considered as influential individual nodes; thus, the problem is how to select the set which can establish a wide advertisement. Like computer viruses in computer networks and epidemic in population and scandals in community, it is required to select influential nodes such that negative and destructive spread is prevented. Graph theory is a branch of mathematics which discusses graphs (Zhang, 2011, 2018); in fact, it is a branch of topology which is closely related to algebra and matrix theory (Nuwagaba and Hui, 2015.).

2 Literature Review

2.1 Independent cascade model

In this model, the whole social network is represented as a graph in which vertices represent individuals and edges represent relationship among nodes. Some of these nodes are active and some of them are inactive. In addition, some of these nodes are not related to other nodes and some are extensively related to other nodes. Weight of edges represents relationship level and dependency of graph nodes. In fact, it describes a step-by-step broadcast procedure and it is concentrated on information transmitter. Information flow starts from primary nodes called seeds which are active at the beginning. When a node (seed) is activated for the first time at time instant t, it finds the possibility at the next step, t+1, to influence its inactive neighbors. This influence is proportional to predefined probability on the link between the two nodes. If the transmitter node succeeds to influence the other, the receiver node receives the information and tries to spread it in the network. It should be noted that whether the transmitter node influences the receiver node or not, the node cannot try to activate the receiver node in subsequent time steps. In the next step, the recently activated node is allowed to spread information in the network. Thus, as time goes on, active nodes of the network are increased. This procedure is continued in discrete time intervals until no active node becomes inactive.

2.2 Linear threshold model

In this model, like independent cascade model, social network is modelled using a graph in which vertices represent individuals and relationships among individuals are represented by edges. On the other hand, each vertex has two active and inactive modes. In this model, in addition to probability of influencing links, another parameter is also defined for each node, which determines influence threshold. Each node V selects its influence threshold randomly from [0, 1]. In this model, like ICM (independent cascade model), spread of influence in the network is repetitious in discrete time steps with this difference that focus is on information receiver node. In fact, this model investigates influence of neighboring nodes. Influence degrees on input links of a node v are selected such that total influence degrees from neighbors of that node do not exceed 1. In this model, a node is activated when total influence degrees received from active neighbors becomes greater than or equal to its influence threshold.

3 Problem Statement

Influence maximization is one of the basic problems in social networks. Viral marketing is one of the applications of this context (word of mouth). A small class of users is selected to accept a product and its word of mouth can result in wide acceptance of this product in social networks. Influence maximization can be defined as: determining a small set of nodes which are able to establish large cascades of behavior which is spread in the whole network. Influence maximization is selection of a set comprising k nodes of the social network such that its spread in the network is maximized. Motivation of the proposed method, C-K-shell, is that in the previous study, K-shell was implemented on the whole network and a long time was spent to find influential nodes. In the proposed approach, SLPA algorithm is used to divide the network to smaller segments and K-shell is applied to each segment so that speed is increased and influential nodes are found faster.

4 The Proposed Method

4.1 Community detection K-shell (C-K-shell) method

Purpose of this method is to increase influence in social networks. Considering increasing growth of social networks for information broadcast and advertisements in social networks, methods should be presented to offer announcements to fewer people and they should broadcast this information to influential individuals and spread information in the whole network. To this end, fast and scalable algorithms should be presented. If a network becomes large and larger, algorithm's speed is not reduced. For example, there are one milliard members in Facebook and broadcasting information among one milliard people is very difficult which requires fast and scalable algorithms.

C-K-shell which is presented in this paper operates based on LT model which is proposed to solve time and number of polluted nodes for increasing influence in social networks. General framework of this method is based on greedy algorithm which finds communities of each data using slap algorithm and uses capabilities of K-shell and community detection to offer a method for maximizing influence in social networks. First, it detects communities, allocates a share to each community, that is, it assigns a share of seeds to each community, and it is decomposed using K-shell algorithm, sorts them in descending order and selects seeds with maximum marginal effect. Using K-shell and community detection algorithms, propagation speed increases and influence maximization can be improved.

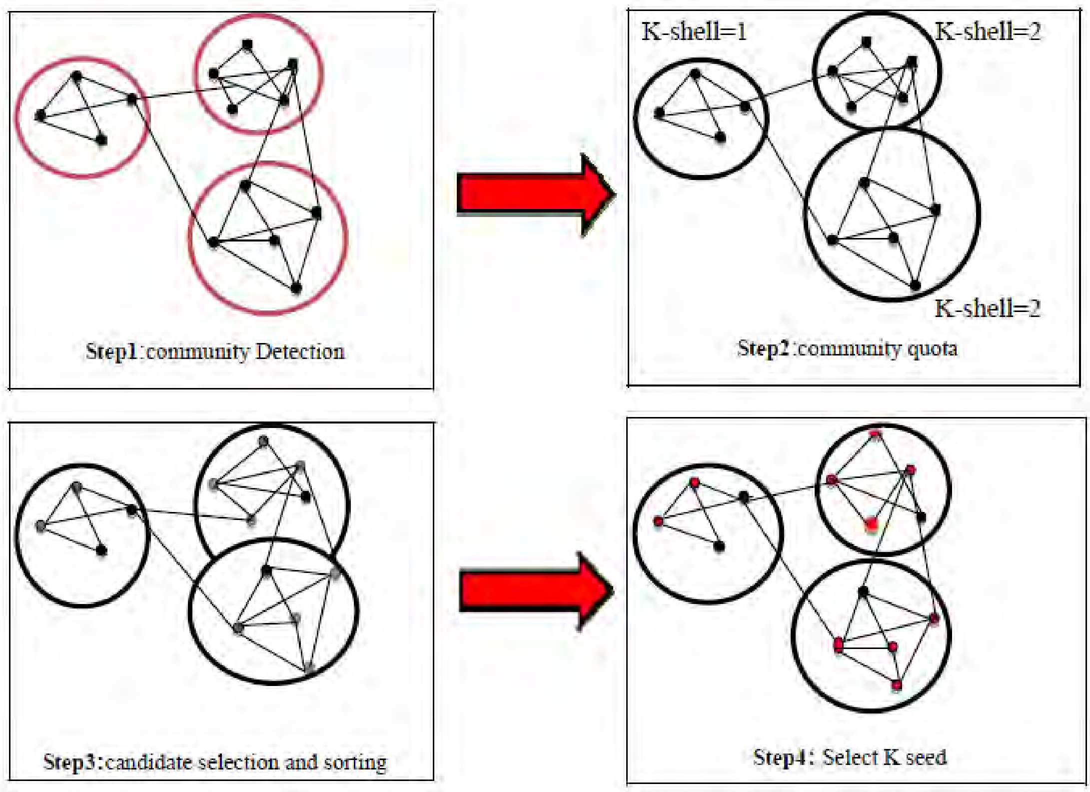

Fig. 1 The steps of proposed algorithm.

4.2 Suggested algorithm

Algorithm 1: C-K-shell (G, k)

Input: Graph G (V, E) and k (number of requested seeds)

Output: k selected seeds

Step 1: Community Detection

CN = Detect communities of G using SLPA algorithm;

Step 2: Community quota compotation

Run k-shall Decomposition Algorithm Each community

Community quota=K-shell*(*community-degree*)/(*community-nodes*)

Step 3:candidate selection and sorting

Sorting inflation nods by marginal

Step 4: Select K seed

Seeds = select k top seed nods

Output k Seeds ;

4.3 Steps of the proposed algorithm

4.3.1 Community detection

In the first step, communities of the input network are selected using SLPA algorithm. Fig. 1 shows simulation of information spread based on mutual influence of auditor. SLPA is used due to its high speed in community detection. This algorithm is the extended version of LPA (Label Propagation Model) method. In this method, each node first has one tag and it is updated continuously to achieve a proper similarity with neighboring nodes and tags. Separate sub graphs are extracted after completing execution steps. In order to involve overlap problem in sub graphs, each node repeats steps of the first procedure to a certain extent. In real procedures, nature of communities in social networks is dynamic and how information is received and how sub-communities are developed, changes continuously. The algorithm used in this section is developed to repeat communication of human factors in real world. In this method, unlike initial version of LPA, information is stored in each node incrementally and removing information in graph section is prevented. Pseudo-codes shows summarized procedure of the algorithm (Fig. 2).

```
Algorithm : SLPA(T, r)
  [n,Nodes]=loadnetwork();
  Stage 1: initialization
  for i = 1 : n do
     Nodes(i).Mem=i;
  Stage 2: evolution
  for t = 1 : T do
     Nodes.ShuffleOrder();
     for i = 1 : n do
        Listener=Nodes(i);
        Speakers=Nodes(i).getNbs();
        for j = 1 : Speakers.len  do
           LabelList(j)= Speakers(j).speakerRule();
        w=Listener.listenerRule(LabelList);
        Listener.Mem.add(w);
  Stage 3: post-processing
  for i = 1 : n do
     remove Nodes(i) labels seen with probability < r;
```

Fig. 2 Pseudo-codes of algorithm SLPA (Xie, 2011).

In general, SLPA has three main steps which are described in the following (Fig. 3):

Step 1:

A memory is created for each node which includes indicator and ID of each node.

Step 2:

The following steps are implemented continuously until the determined number of iterations is reached.

- A node is selected as auditor.
- All neighbors transmit a packet to the auditor as speaker. Iterations and probability of exchanging packets is stored in memory of auditor nodes.
- Auditor node only accepts one of the tags received from its neighbors involving role of auditor. The selected tag is selected as the most popular tag among available information.

Step 3:

In the last step, post-processing is performed. Based on iterations of available tags in memory of each node, tag with maximum iteration is selected as tag of the community.

Fig. 3 Converting the graph to the sub-graph.

In the following, graphical steps of implementing the graph in SLPA is presented (Fig. 4).

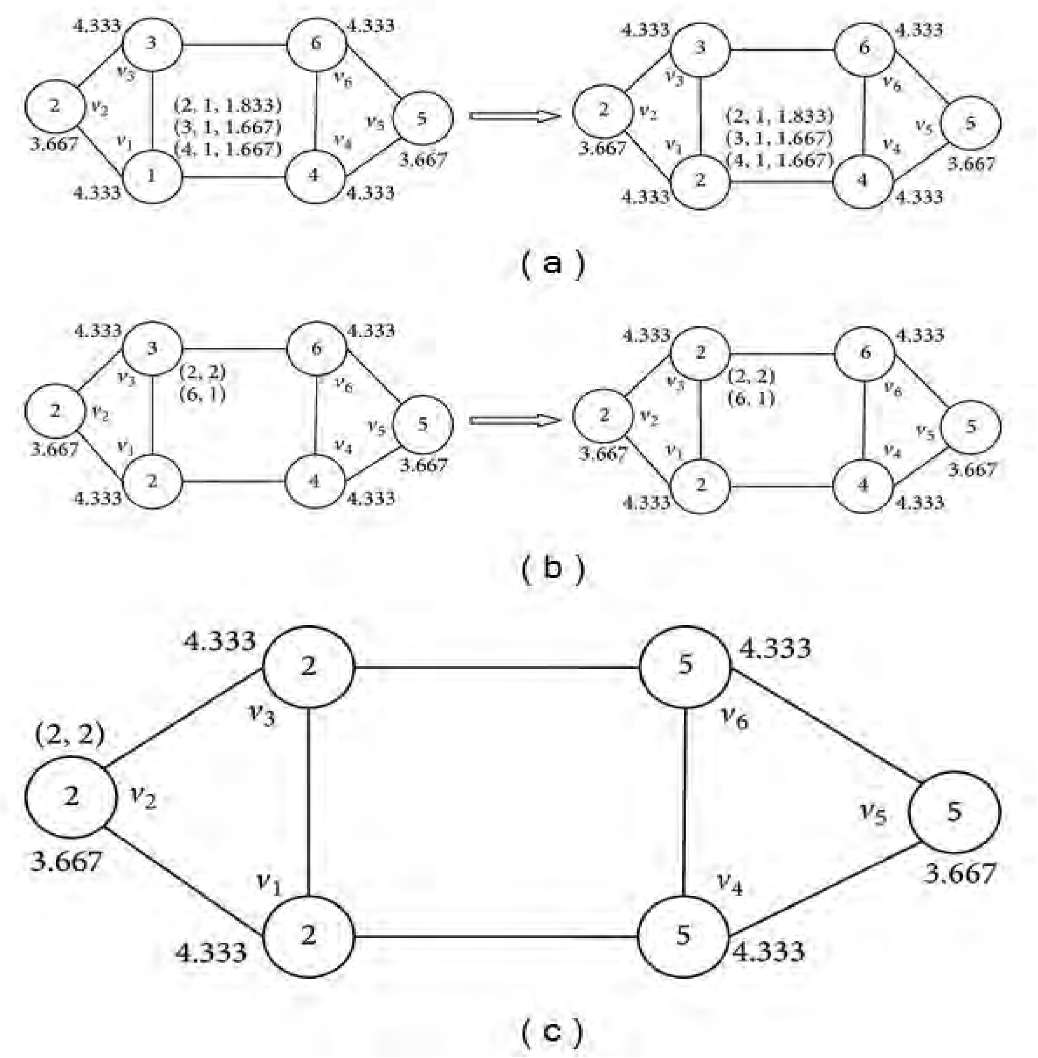

Fig. 4 Execution steps of the algorithm.

In order to analyze communities, some codes are presented to develop a modified version of the dynamic algorithm.

4.3.2 K-shell decomposition and allocating share to each community based on number of nodes of each community

In this step, second K-shell decomposition form is performed. K-shell decomposition is a method which ranks influential nodes with high efficiency in complex networks. However, its disadvantage is repetition and creating generated information while ranking influential nodes. This method is a new and fast ranking method for evaluating influence of each node; the assessment parameter of the proposed approach is iteration factor used in K-shell approach. Empirical results show that the proposed method outperforms other methods in terms of accuracy, efficiency and uniformity both in simulation and implementation. This approach shows that nodes with higher influence and accuracy are ranked in a separate list.

According to the second Fig. 4, when communities are detected using SLPA algorithm and K-shell is performed on communities, a share is assigned to detected communities based on nodes of each community. For instance, 3 shares are given to the first community, 1 share is given to the second community; thus, a share is allocated to each community. As shown in the Fig. 4, K-shell of some nodes is higher and K-shell of other nodes is smaller because nodes with higher K-shell have higher importance and higher share is allocated to them; no share is allocated to nodes with smaller K-shell; shares are increased and decreased based on K-shell.

Community quota=K-shell*(*community-degree*)/(*community-nodes*)

In this equation, in order to share K-shell communities, each node multiplied by number of edges externally connected to the community divided by number of nodes of each community is used; thus, a share of seeds is allocated to each community. Purpose of C-K-shell is to increase speed and detect influential nodes faster by dividing the network to smaller sections and performing K-shell on each section.

4.3.3 Selecting candidate nodes and sorting them in descending order

In this step, according to Fig. 4, after K-shell decomposition and allocating share to each community, candidate nodes are sorted. In this step, number of candidate nodes of each community is two times the share of each community. It is clear that in communities, the nodes which are closer to network center and have higher marginal effect are more important and more influential. After selecting candidate nodes, they are sorted in descending order.

4.3.4 Selecting seeds

In this step, k initial seeds are selected from list of important nodes such that they have maximum influence in the input network. Some of the superior nodes are selected from the sorted list based on their share and k top nodes which have a high rank are selected as the final k set.

5 Related Studies

Pittel et al. (1996) proposed a K-shell decomposition method to evaluate potential influence of nodes in which all nodes with degree 1 and connected edges are eliminated. When nodes with degree 1 are eliminated, some of the nodes might become single-edge or their degree becomes 1; thus, this procedure is continued until there is no single-edge node. The removed single-edge nodes are called shell-1. The same procedure is done for two-edge nodes which are called shell-2 and the procedure is continued for nodes of degree 3 and 4 until the same score or K-shell is assigned to the nodes (Pittel et al., 1996).

Kempe et al. (2003) proposed greedy algorithm in 2003 which can ensure quality of finding the optimal graph structure. But the algorithm is very time-consuming. Therefore, Linear Threshold and Independent Cascade models have recently attracted attentions for finding influential nodes in social networks (Wang, 2015).

Cost Efficient Lay Forward (CELF) was proposed by Leskovec et al. (2007), which is based on using influence extension function which is a submodular function; it increases different nodes to the current set S which adds the same element to a set better than S. Therefore, CELF decreases number of estimation extension function recall achieved through not employing similar recalls. Greedy algorithms have low execution time and its computations are performed in memory. However, time complexity of CELF in the first iteration is high and requires a long time for initialization (Zhou et al., 2017).

In the heuristic method presented by Chen et al. (2010), a DAG is built for each node V, nodes are extended locally, DAG (Directed Acyclic Graph) is calculated and the basic node is selected based on the greedy algorithm. Execution time of this algorithm is acceptable but since it does not employ a proper storage structure, a large amount of memory is used to store a DAG for each node. Main idea of the proposed algorithm is to create local trees with starting or ending nodes and edges at each node; probability of activating a node by other nodes is calculated in the foundation of neighboring nodes. Finally, one or several nodes with maximum probability are presented as influential nodes (Mihara, 2015; Gong, 2016).

Kitsak et al. (2010) showed that most influential nodes are the ones concentrated in the network core known as K-shell decomposition; using this method, the network is decomposed to k shells and each node receives a K-shell value after decomposition which is based on its position. The closer it is to center of the networks, its score and K-shell would be higher.

Guyal et al. (2011) proposed SIMPATH algorithm to decrease number of recalls. First, nodes of the graph axis are guessed and investigated in different iterations of neighboring nodes. Knowing extension values, basic nodes in K iterations are found. Yu et al. (2013) found influence route for each node V starting from V towards other nodes and calculating influence probability for each node. Node with maximum influence is selected as the main and effective node. The proposed algorithm has a proper and linear execution time and maximizes efficiency by parallelizing the multi-core system. Main disadvantage of this method is that influential active node with low quality is compared with other algorithms more than other datasets (Tsai, 2015; Sankar, 2016).

In 2016, Bagheri et al. (2016) proposed a fast and efficient algorithm for influence maximization in social networks called COMPATH since previous algorithms were not scalable and required long time in large networks. This algorithm investigates a few number of nodes, first the communities existing in social networks are detected and then seeds of important communities are obtained using COMPATH with this difference that a weight is considered for each edge. The related weight shows importance of nodes compared to each other. Empirical results show that their method gives better results compared to COMPATH and other methods. Quality of seeds show that this algorithm is very faster and more efficient than current algorithms (Bagheri et al., 2016).

In 2017, Bagheri et al. proposed a method to maximize influence called FSIM. First, slap is used to detect communities, then each community is considered as a node. Then, a new network is created using detected communities. Then, candidate nodes of each community are selected and finally the final seed is selected among the candidate nodes. This algorithm can investigate nodes without loss of quality, find seeds quickly and maximize influence in social networks. Empirical results show that FSIM is faster and more scalable than other algorithms (Bagheri et al., 2017).

The problem of optimization and optimization of the work has been presented in 2003 by Kempe et al., and they presented a greedy algorithm of GA by approximation, which can provide an optimal result of maximum effect. The algorithm iteratively selects nodes with the highest degree of the marginal effect, adding them to the K - node set and therefore, due to the time - consuming and not optimal solution, the overall solution is not appropriate. On the other hand, there is another algorithm that is very simple and very simple and has low computational cost, but because it has an unstable accuracy and is unreliable, we have used the advantage of greedy algorithms and new algorithms to solve these problems, and we have combined them together and presented a KDA algorithm (Kempe et al., 2003).

Estevez et al. (2007) presented a method based on a greedy model to maximize the impact on the graph. They discussed and analyzed two main models in their work, including the linear threshold model as well as the independent development model. The methods presented in this study were analyzed and analyzed on the complex graph graphs of social networks. This method is one of the most suitable algorithms in terms of running time and has a linear running time but due to greedy approaches it has lower performance and in many cases ineffective or lower impact coefficients are selected.

Goa et al. (2013) investigated the problem of maximizing penetration from a new perspective. First, the K node is selected as the assumed base and effective set, in which the maximum influence on a set of target nodes is considered to obtain a subset of effective nodes and high penetration communities. Infiltration coefficients under societies in which base nodes is examined from a public social network for characteristic index analysis (Chen, 2014).

Kim et al. proposed a method in 2013 which includes selecting communities from the graph by a mixed - Markov chain approach. In each society a node is selected as an influential and influential node candidate. As a result, K nodes are selected by extending greater diffusion from candidate nodes and frequently have all the conditions of the effective node in those societies. The main problem of this algorithm is the lack of attention to the role of each community as a unit of influence and size of each community (Ni, 2017).

Cheng et al. (2014) prioritize all the different segments of the input graph and then analyze the communities algorithm to update these rankings in different iterations. Then, based on this initial rank, the node expansion is estimated. In the greedy algorithm, this is mainly a region-based study that leads to the same inefficiency.

In 2015, Wang et al. reviewed the issue of maximizing the impact of social media communications. In his thesis, he discussed two main issues. The first issue is the minimum number of communication links in which the goal is to find the minimum number of communication links that have the most impact on the entire network. The second problem is to maximize the impact on virtual networks, where the goal is to find a certain number of neighbors so that a set is selected that has the most impact on the original graph. They also investigated these two problems in different network states, such as rings, trees, and in-tree subgraphs, and proposed an algorithm for polynomial execution and suitable for solving these two problems. Proper execution times and inefficiencies for the filler and bulk graphs have been the features of the algorithm in question.

One of the major issues in this domain and the algorithms presented in the current domain is the high time run of the algorithms due to the complex nature of these types of problems and algorithms. Therefore, the optimal, economical and cost-effective choice of primary users to maximize the impact factor has been introduced. Applying a basic ranking model based on communication and its levels is a well-executed algorithm for analyzing such complex graphs. This algorithm has the proper time efficiency of polynomial degrees and the algorithm has difficulty finding sub-efficient spanning communities. This is a good method if you use graphs with a centrality (Chen et al., 2010).

Guo et al. (2013) presented a method based on the discrete particle optimization model based approach to maximize the impact on different social network graphs in 2016. They also utilized their defined evaluation function on this dataset to improve the performance of the algorithm, which provides higher performance than other algorithms for investigating such graphs. Particle optimization algorithms and general evolutionary algorithms require appropriate evaluation function and optimal parameter determination. This feature reduces system runtime and system performance and makes it difficult to use for high-level problems.

Liu et al. (2017) proposed a method based on priority selection and heuristics which are used to optimize the impact on virtual networks, and specifically the real models of these networks, which are dynamically changing. They were analyzed on Facebook databases, and Flickr was analyzed. This algorithm has a relatively convenient execution time but due to the lack of high - volume memory, there is no good performance in the issues of social networks as well as the lack of proper crack failure and local efficiency of parts of the algorithm is not appropriate for analysis of different graph problems.

Zhu et al. presented a comprehensive analysis of the use of an algorithm to optimize the performance of current algorithms. They analyzed this type of algorithms in different manufacturing markets and showed that using data from this type of systems, the production and effect of different parts can be optimized if there is a more system for efficiency. one of the main issues in their analysis is the dynamic nature of the issue that is constantly changing (Kim et al., 2013).

Among proposed approaches, fuzzy modeling has used fuzzy modeling to maximize the influence factor in social networks and model based on the use of destination nodes. They also compared their proposed method in 2017 and tested and evaluated their results (Cheng et al., 2014).

6 Analysis of Empirical Results

The method proposed in this paper is implemented using visual studio 2015. The following Table 1 shows characteristics of the system on which the proposed method is implemented.

Table 1 System specification for simulation and evaluation of results.

Specification	Hardware / software
Windows 7	Operating system
Operating system64bit	Operating system type
4GB – 3.06GB usable	MemoryRAM
Cpu:Intel® Core™2 Duo,2M cache, BUS 800MHz	Processor

6.1 Datasets

In this section, datasets studied in this paper are introduced. In order to evaluate the proposed solution, studies are performed on two real social networks including Epinions and NetHEPT.

6.2 Epinions network

Epinions is a site in which users can review advertisement items and score them. In this network, trust among users and scores assigned to items are combined to show specific items to each user and it is expected that higher score is assigned to that item. This large dataset includes 75000 nodes and 500000 edges.

6.3 Nethept network

This is a network comprised of cooperation among a set of networks adopted from “high energy physics theory” which has 15233 nodes indicating authors and 58891 edges representing cooperation among authors. This dataset is widely used to assess influence maximization.

6.4 Evaluating efficiency of the algorithm

The proposed algorithm employs K-shell to calculate influence of each node. The current algorithm offers better efficiency compared to CIM, PaS, CGA and other methods like ComPath. This algorithm not only offers better efficiency but also it has better execution time. An algorithm has high efficiency and quality when it has high spread of influence. The proposed approach is compared with other methods and its results are given in the following. Results of spread of influence are given for different number of seeds for input data of Epinions and NetHEPT. In the following, assessment diagrams for two datasets studied in this paper are investigated to compare quality of influential nodes.

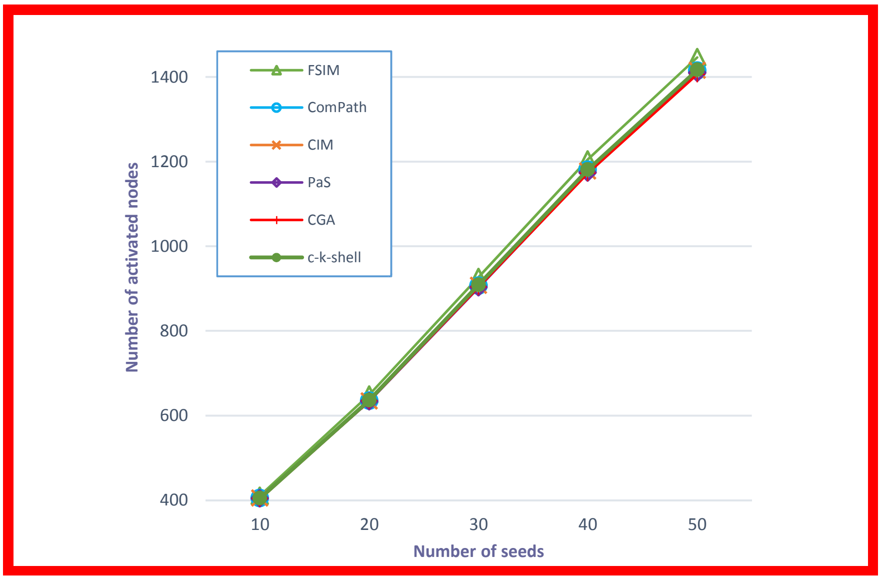

Fig. 5 Number of activated nodes by the algorithm used and comparing it to the other five methods on NetHEPT data.

As shown in the Fig. 5, as number of seeds increases, spread of influence also increases. On the other hand, performance of the proposed approach is improved as number of seeds increases. As number of seeds increases, system performance is also improved and offers better output compared to other methods except FSIM. In the following, results of applying the proposed approach on input data of NetHEPT are studied.

Results of the proposed algorithm are more suitable and optimal than other methods. Results of the proposed approach are compared with 5 other methods. Results of NetHEPT are similar to the previous case. As number of seeds increases, the proposed algorithm performs better than other methods. Thus, using this approach might be better. This procedure is investigated using Epinions data (Fig. 6).

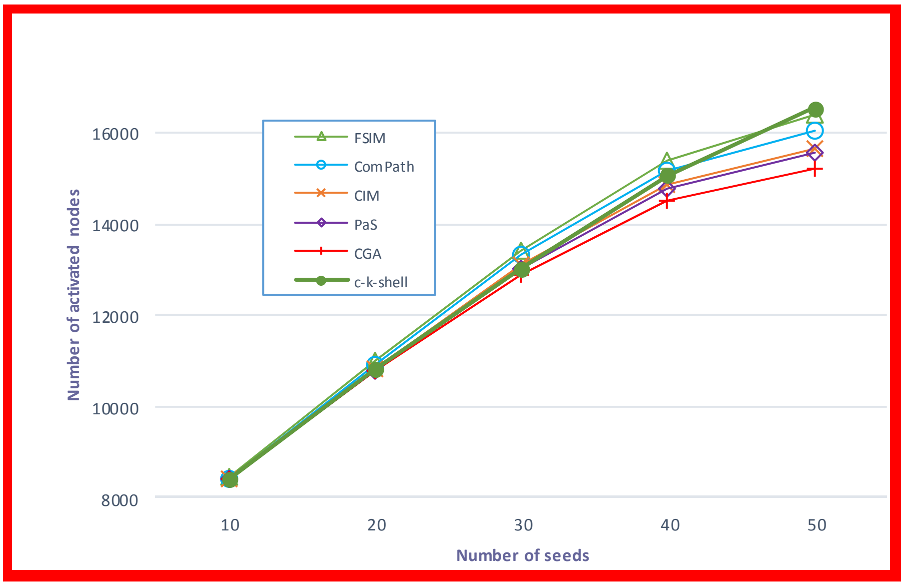

Fig. 6 Number of activated nodes by the algorithm used and its comparison on the other five methods on the Epinions data.

It is observed that results of Epinions are similar to previous case and efficiency of the proposed method can be seen obviously.

Comparing execution time of the compared algorithms for two presented datasets, the following results are obtained. According to the results, the proposed algorithm has the best execution time and can compete with other algorithms in terms of execution time (Fig. 7-8).

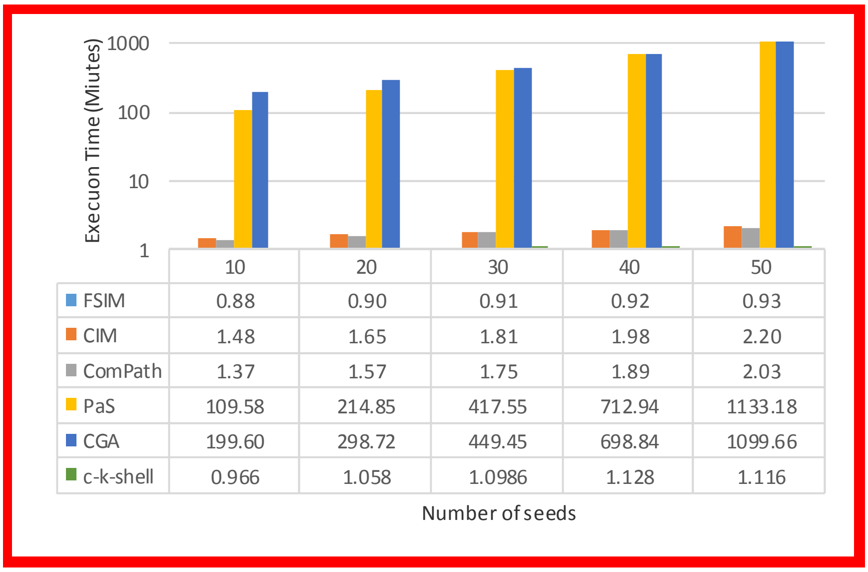

Fig. 7 Comparison of the execution time of different algorithms per seed number per NetHEPT dataset.

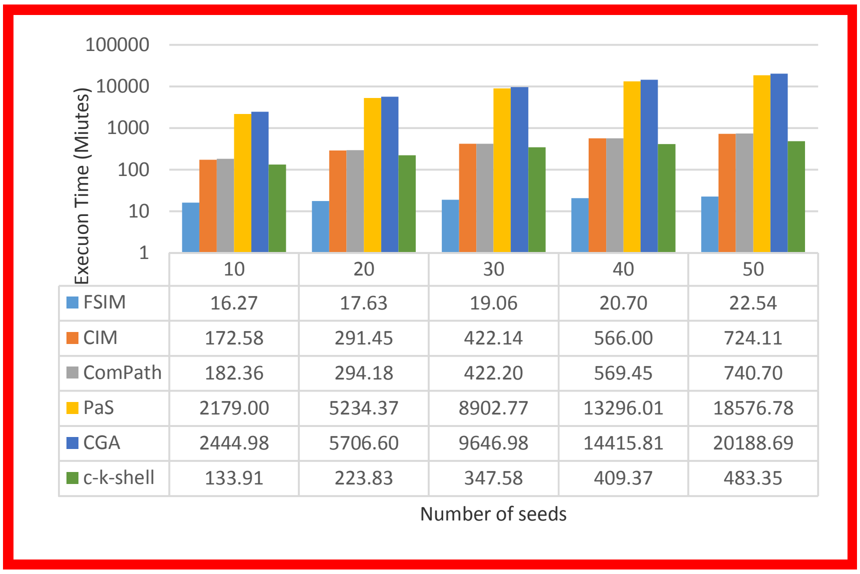

Fig. 8 Comparison of execution time of different algorithms per seed number per Epinions dataset.

Since the proposed algorithm performs better than other methods, it is compared with 5 other methods in terms of execution time. Since the proposed method outperforms other methods, it is shown that it offers acceptable performance in terms of execution time. As can be seen in Fig. 7-8, CGA performs weaker than other methods.

CIM, PaS, ComPath and CGA are implemented 25 times on Epinions and NetHEPT and their average is calculated. As discussed in the previous sections, the proposed algorithm performs better than 4 other methods in terms of number of active nodes and execution time; but it performs approximately the same as FSIM in terms of execution time and number of active nodes. But the proposed method has shown proper performance in terms of execution time compared to CIM, PaS, ComPath and CGA. But it performs almost the same as FSIM.

7 Conclusion

In this paper, a fast and scalable algorithm is presented to maximize influence (C-K-shell) based on community detection networks. C-K-shell reduces number of nodes which should be investigated to find seeds without losing quality. C-K-shell uses SLPA to detect communities from input network and develop a new network. The new network has two nodes in which each node represents a community. Therefore, only a limited number of nodes are investigated so that speed is increased. Important communities are selected based on their connection strength in new networks. Important nodes of important communities are selected. Initial seeds are selected and after testing initial seeds, important nodes and final seeds are selected. Empirical results show that C-K-shell is the fastest algorithm compared to algorithms which can select high quality seeds. It is suggested to use different scales like close centrality, betweeness centrality and dynamic method for detecting communities and determining share of community (Shams and Khansari, 2014; Byron and Tennenhouse, 2015; Zhang, 2015; Khansari et al., 2016).

References

Bagheri E, Dastghaibyfard G, Hamzeh A. 2016. An efficient and fast influence maximization algorithm based on community detection. 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery. 1636-1641

Byron CJ, Tennenhouse C. 2015. Commonality in structure among food web networks. Network Biology, 2015, 5(4): 146-162

Chen W, et al. 2014. Realtime topic-aware influence maximization using preprocessing. Computational Social Networks, 3: 8

Chen W, Wang C, Wang Y. 2010. Scalable influence maximization for prevalent viral marketing in large-scale social networks. Proceedings of SIGKDD. 1029-1038

Cheng S, Shen HW, Huang J, Chen W. 2014. Imrank: Influence maximization via finding self-consistent ranking. Proceedings of SIGIR, 475-484

Domingos P, Richardson M. 2001. Mining the network value of customers. Proceedings of the Seventh ACM and SIGKDD. International Conference on Knowledge Discovery and Data Mining. 57-66

Estevez PA, et al. 2007. Selecting the most influential nodes in social networks. International Joint Conference on Neural Networks. IEEE, 2397-2402

Gong MG. 2016. Influence maximization in social networks based on discrete particle swarm optimization. International Journal of Information Sciences, 614(C): 367-600

Guo J, Zhang P, Zhou C, Cao Y. 2013. Personalized influence maximization on social networks. Proceedings of CIKM. 199-208

Kempe D, Kleinberg J, Tardos E. 2003. Maximizing the spread of influence through a social network. Proceedings of the Ninth ACM SIGKDD. International Conference on Knowledge Discovery and Data Mining.

Khansari M, Kaveh A, Heshmati Z, et al. 2016. Centrality measures for immunization of weighted networks.

Network Biology, 6(1): 12-27
Kim C, Lee S, Park S , Lee S. 2013. Influence maximization algorithm using Markov clustering. Database Systems for Advanced Applications, 112-116
Kitsak M, et al. 2010. Identification of influential spreaders in complex networks. Nature Physics, 6(11): 888-893
Leskovec J, Krause A, Guestrin C, Faloutsos C, et al. 2007. Cost-effective outbreak detection in networks. Proceedings of SIGKDD. 420-429
Liu XD, et al. 2017. On the shoulders of giants, incremental influence maximization in evolving social netoworks. Complexity, 2017: 5049836
Mihara S. 2015. Influence maximization problem for unknows social networks. IEEE/ACM International Conference on Advances in Social networks analysis and Mining.
Ni Y. 2017. Optimization influence diffusion in a social network with fuzzy costs for targeting nodes. Journal of Ambient Intelligence and Harmonized Computing, 8(5)
Nuwagaba S, Hui C. 2015. The architecture of antagonistic networks: Node degree distribution, compartmentalization and nestedness. Computational Ecology and Software, 5(4): 317-327
Pittel B, Spencer J, Wormald N. 1996. Sudden emergence of a giant k-core in a random graph. Journal of Combinatorial Theory, Series B 67: 111-151
Sankar C, eet al. 2016. Learning from bees: An approach for influence maximization on ViralCompaigns, Plos one, 0168125
Shams B, Khansari M. 2014. Using network properties to evaluate targeted immunization algorithms. Network Biology, 4(3): 74-94
Tsai CW, et al. 2015. A genetic new greedy algorithm for influence maximization in social networks. IEEE International Conference on Systems, Man and Cybernetics.
Wang YF. 2015. PPRank: economically selecting initial users for influence maximization in social networks. IEEE Systems Journal, 99
Xie JR. 2001. Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process. IEEE International Conference on Data Mining Workshops.
Zhang WJ. 2011. Network Biology: an exciting frontier science. Network Biology, 1(1): 79-80
Zhang WJ. 2015. A generalized network evolution model and self-organization theory on community assembly. Selforganizology, 2(3): 55-64
Zhang WJ. 2018. Fundamentals of Network Biology. World Scientific Europe, London, UK
Zhou JY, Fan JX, Cheng BL. 2017. Cost-efficient influence maximization in onlince social networks. The Fifth International Conference on Advanced Cloud and Big Data. 64-71