An influence maximization algorithm based on community detection using topological features

2021 11th International Conference on Computer Engineering and Knowledge (ICCKE) | 978-1-6654-0208-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICCKE54056.2021.9721514

Zahra Aghaee
Department of Software Engineering
Faculty of Computer Engineering
Isfahan, Iran
Z.Aghaee@eng.ui.ac.ir

Afsaneh Fatemi
Department of Software Engineering
Faculty of Computer Engineering
Isfahan, Iran
a_fatemi@eng.ui.ac.ir

***Abstract*—Due to the growning use of social networks and the use of viral marketing in these networks, finding influential people to maximize information diffusion is considered. This problem is Influence Maximization Problem on social networks. The main goal of this Problem is to select a set of influential nodes to maximize the influence spread in a social network. Researchers in this field have proposed different algorithms, but finding the influential people in the shortest possible time is still a challenge that has attracted the attention of researchers. Therefore, in this paper, the IMPT-C algorithm is presented with a focus on graph pre-processing in order to reduce the search space based on community structure. The approach of this algorithm is to take advantage of the topological properties of the graph to identify influential nodes. The experiment results indicate that the IMPT-C algorithm has a great influence spread with low run time compared the state-of-the-art algorithms consist least 2.36% improve than PHG in term the influence spread.**

***Keywords*—social network; viral marketing; Influence Maximization Problem; information diffusion; community detection.**

I. INTRODUCTION

Today, people use various Social Networks (SNs) such as Facebook, Twitter, LinkedIn, and Instagram to communicate with friends, obtain information in various fields, education, sharing ideas and opinions, spreading news and rumors, marketing and selling products. Various information is quickly transmitted from person to person through the messages of individuals with each other on SNs. This method of information diffusion is called word-of-mouth [1, 2]. For example, people start promoting their product to influence a huge number of people in a SN, which is marketing at the level of a SN. This type of advertising is known as viral marketing [3]. The important problem in this type of marketing is finding influential people in a SN. It gives the most Influence Spread (IS) in the shortest possible time that is called Influence Maximization Problem (IMP) on SN [4, 5].

In the IMP, one of the important factors is initially activated individuals that the diffusion process from them begins, which are called seed nodes [1, 6]. Another factor is the number of individuals activated at the end of the diffusion called the IS. So, it examines individuals activated by seed nodes under a diffusion model [7]. Another factor is the time elapsed from the beginning to the end of the diffusion process [7]. In general, the main goal in IMP is to find influential people in a SN and diffusion in the shortest possible time [8]. Accordingly, in the SN graph $G\ (V, E)$, as shown in Equation 1 [9], the S subset should be selected as seed set from the V nodes in such a way that under a diffusion model with probability p leading to activating most the number in $G\ (V, E)$. Of course, the process of selecting seed set is repeated until $|\ S\ | \ = \ k$.

$$S^* = argmax_{S \subset V, |S|=k} \sigma(S) \qquad (1)$$

In Equation (1), S^* represents k nodes with the highest IS in $G\ (V, E)$.

Domingos and Richardson first proposed IMP as an algorithmic approach to probability [10]. Then, Kempe et al. Formulated this as a discrete stochastic optimization problem [8]. They proved that the IMP is NP-hard [8]. Then, to solve it, they presented a General Greedy (GG) with an optimal approximation of 1-1 / e≃0.63123 [7, 8]. It has a high IS, but the disadvantage of that very high run time is due to the number of repetitions of the Monte Carlo (MC) simulation; Therefore, due to the increasing use of SNs and the increasing networks size, the GG is not suitable for these networks. Then various other algorithms in addition to greedy algorithms [7, 11, 12], including heuristic algorithms [7, 13-17], community-based algorithms [4, 18, 19], and meta-heuristic algorithms [20-23] were presented by researchers with the goal of maximizing the IS.

The algorithms proposed in recent years need to be improved in several respects: the study of core nodes and their effect on the IS and pre-processing to reduce computational overhead. Therefore, in this paper, a technique is proposed to solve the IMP based on the Topology and Community detection (IMPT-C) algorithm, to improve IS and run time efficiency. In IMPT-C, Candidate Nodes (CNs) are identified using a kind of graph pre-processing based on weighting different communities. Then, the influential nodes are selected from the CNs based on the topological characteristics of the graph and using Best First Search (BFS). Experiments indicate that the IMPT-C has given better results in terms of IS and run time than the state-of-the-art algorithms on different data sets based on the Independent Cascade (IC) diffusion model.

In summary, the contribution of this paper is providing IMPT-C algorithm with a improve approach that reduces selective space to detect influential nodes, reduces

computational overhead, and thus improves time-efficiency. The proposed algorithm investigates the position of core nodes and their effect on the graph's global propagation according to the graph's topological properties. The IMPT-C examines the effect of the community detection to detect the seed set.

The sections of the paper are: Section 2 explains related algorithms. Next, Section 3 describes the IMPT-C algorithm in detail. Section 4 also evaluates and presents the proposed algorithm's results compared with other algorithms on the different data sets. Also, Section 5 indicates conclusion.

II. Related Works

The IMP has not an exact solution in polynomial time [8]. Kempe et al. First introduced the GG algorithm in 2003 [8] that the higher iterations of the MC gives the better the IS in terms of accuracy [8]. Then, due to the high run time and lack of scalability of the GG algorithm, the researchers presented algorithms that can solve its drawbacks. TABLE I shows the various algorithms proposed for the IMP on SNs along with their attributes. In this table, the general information of the IMP algorithm, including the information diffusion model used, the category of each algorithm, their advantages and disadvantages are shown. Also, "✓" means "used" and "✗" means "not used".

III. Proposed Algorithm

The proposed IMPT-C algorithm is a community-based method, which is presented using graph topology features. In IMPT-C, using the initial pre-processing on the graph,to reduce the selective space to detect influential nodes, the CNs are selected. Then the seed set are chosen using the topology features of the graph and using the approach of the data structure of the BFS. This causes the nodes to be evaluated with different features, and finally, the node with the highest IS is chosen as the seed node. The flowchart of the proposed IMPT-C algorithm is indicated in Fig. 1. Generally, this algorithm includes of three basic phases. These phases are: 1- Pre-processing and selection of CNs, 2- Calculate the importance of CNs based on the topology features of the graph, and 3- Selecte the seed set. In the following, the phases of the proposed algorithm are described in detail.

A. Pre-processing and selection of CNs

In this phase of the proposed IMPT-C method, CNs are selected from all graph nodes in order to reduce the selective space for detecting influential nodes. It reduses the computational overhead, and also improves the run time. For this purpose, first, using the Louvain algorithm [24], different communities are identified in the graph. Louvain is a hierarchical clustering algorithm and has a high speed for finding communities. Next, the weight of each community is calculated based on Equation (2), which includes three basic criteria: a) the average degree, b) the density of each community, and 3) the effect of the nodes located in the inner shells of the graph. This phase identifies communities with the probability for optimal IS.

$$W_C = AD + (Dens \times C) \tag{2}$$

In Equation (2), W_C is the weight of each community. The AD criterion is the average sum of the degrees of the nodes of each community relative to the nodes of the whole graph, which is calculated according to Equation (3). The $Dens$ criterion also indicates the density in each of the communities, which is calculated according to Equation (4). The C criterion also deals with the graph topology using the K-shell decomposition [25] criterion and expresses the importance of the nodes in the core; because a node with even the smallest connection in the core may have a significant IS. Therefore, in the graph topology, the position of a node is defined under the K-shell decomposition criterion. This causes nodes with a lower degree and lower diffusion to be at the outer shells and nodes with higher degree and higher diffusion to be at the core of the shelling. Accordingly, Equation (5) is computed for communities whose nodes are located in θ of the inner shell.

$$AD = \frac{\sum_{i=1}^{n_c} d_i}{n_G} \tag{3}$$

In Equation (3), $d_i$is the degree of each node i, n_c is the number of nodes in each community, and n_G is the number of nodes in the whole G.

$$Dens = \frac{2e}{n_c(n_c - 1)} \tag{4}$$

In Equation (4), e is the number of edges within each community.

$$C = \sum_{i=1}^{k_{s_C}} v_c \times k_{s_c} + (PR)_{v_i} \tag{5}$$

In Equation (5), v_c is the number of nodes with shell number k_{s_c} and $(PR)_{v_i}$ is the PageRank [26] value for each CN.

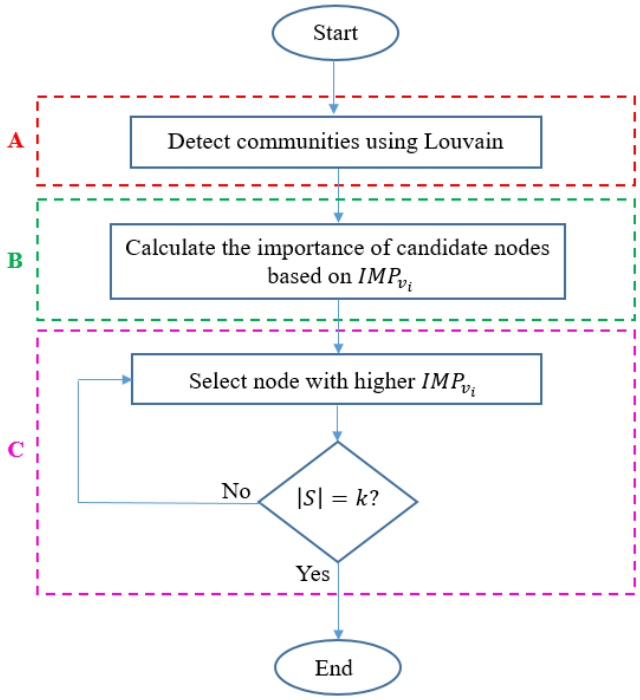


Fig. 1. IMPT-C flowchart

TABLE I. ALGORITHMS PRESENTED FOR THE IMP ON SNS

Algorithm name	Diffusion model			Type	Description
	IC	WC	LT		
General Greedy [8]	✓	✓	✓	Greedy	This algorithm has high accuracy and optimal but has high run time.
CELF [11]	✓	✗	✗	Greedy	This algorithm reduces IS evaluation. CELF run time is low than the GG but not suitable for large-scale graphs.
DegreeDiscount [7]	✓	✓	✓	Heuristic	It reduces degree based on seed set. Degree Discount run time is lower than CELF but does not have optimal approximate.
SAEDV [20]	✓	✗	✗	Meta-heuristic	SAEDV selects influential nodes with the highest Expected Viffusion Value (EDV) using the Simulated Annealing (SA). The SA has a higher IS and lower speed than the DegreeDiscount.
CI [27]	✓	✗	✗	Heuristic	It identifies the seed set by optimal percolation. CI does not have optimal approximate. The result of the CI depends on a circle with radius *L*.
CoFIM [28]	✗	✓	✗	Community-based	It calculates the IS by diffusion among communities. Run time in the The CoFIM is low, But the overlapping nodes are not selected.
LIR [13]	✓	✗	✓	Heuristic	It choses the seed set by the Local Index Rank. Run time in the The LIR is low, but does not have optimal approximate.
ProbDegree [29]	✓	✗	✗	Heuristic	It uses the effect of multi-hop neighbors. Run time in the The ProbDegree is low, but the seed noes in this algorithm does not have optimal IS.
DDSE [21]	✓	✗	✗	Meta-heuristic	This algorithm selects seed set based on Genetic. The DDSE has IS close to the CELF on some small graph. The Run time in this algorithm is more related to local search.
PHG [18]	✗	✗	✓	Community-based	It identifies the seed set by the weight of nodes' influence and community properties. Computational parameters in PHG are important in the run time.
GWIM [23]	✓	✗	✗	Meta-heuristic	This algorithm is based on the grey wolf. The GWIM has a higher IS and lower run time than the DDSE but does not use discretization rules.
TI-SC [4]	✓	✗	✗	Community-based	It uses core node in community for selecting seed set. Run time is related to the number of communities in TI-SC.
GIN [14]	✓	✗	✗	Heuristic	it makes groups of graph with higher connections. The GIN has the IS vary close to HighDegree in some large-scale graphs.
HEDVGreedy [5]	✓	✓	✗	Greedy	It calculates the EDV of the high degree nodes as seed set. The HEDVGreedy has a higher run time than the LIR.
SRFM [30]	✓	✗	✗	Heuristic	SRFM selects seed set by the hierarchical selection. Therefore, this algorithm is very fast, and also, this algorithm has an optimal IS.

Then, in order to reduce the selective space, the communities are sorted based on the weight obtained, and then γ is selected from among them. The existing nodes in these communities are candidate. Therefore, Algorithm 1 is introduced for the first phase of the IMPT-C algorithm.

B. Calculate the importance of CNs

In this phase of the proposed IMPT-C, for each node selected in the previous phase, the importance of that node is identified according to the topology features of the graph. Accordingly, Equation (6) is calculated for each CN v_i.

$$IMP_{v_i} = (d_{v_i} \times \sum_{i \in n_{B_L}} d_i \times p_i) \quad (6)$$

In Equation (6), v_i is the CNs, d_{v_i} is the degree of each CN, L is the BFS depth starting from each CN, and n_{B_L} is the set of the BFS nodes, which are also CNs. Also, d_i is the degree of each node in the set of the BFS nodes, p_i (random number between zero and one) is the probability that v_i will influence d_i.

C. Selecte the seed set

In this phase of the proposed IMPT-C algorithm, the nodes are sorted from highest IMP_{v_i} to lowest IMP_{v_i}. Nexr, k nodes are choosen as seed set. The final seed set are choosen by Algorithm 2.

Algorithm 1: Pre-processing and select CNs

Input: $G\ (V, E)$
Output: CNs
1: initialize $candidate\ node \longrightarrow \emptyset$
2: **for each** c_i in C **do**
3: $W_{c_i} = AD + (Dens \times C)$
4: **end for**
5: choose the number of γ nodes with W_{c_i} largest and add to CNs
6: return CNs

Algorithm 2: Calculate the importance of CNs and select seed set

Input: CNs
Output: Seed set
1: **for each** v_i in CNs **do**
2: $IMP_{v_i} = (d_{v_i} \times \sum_{i \in n_{B_L}} d_i \times p_i)$
3: **end for**
4: Rank the values IMP_{v_i}
5: choose the k nodes with IMP_{v_i} largest and add to seed set
6: return $seed\ set$

IV. EXPERIMENTS

The IMPT-C is tested on various real-world and artificial data sets under the IC model [6, 7, 31]. Then, the results of the algorithm are compared with the results of state-of-the-art algorithms in the Python programming environment on a Windows 10 operating system with a 26 GHz processor (Intel (Core (TN) i5-3230M) and 16 GB of similar RAM in terms of IS and run time. The IS indicates the number of nodes activated during the diffusion. The run time indicates the time taken to select the influential nodes at the end of the algorithm. It is noteworthy that in the proposed IMPT-C algorithm, the parameters of inner shells number $\theta = 3$, number of selected communities $\gamma = 5$, BFS depth $L = 2$, and diffusion probability $p = 0.01$ are used. Also, The IMPT algorithm is the IMPT-C algorithm without phase 1.

A. Data sets

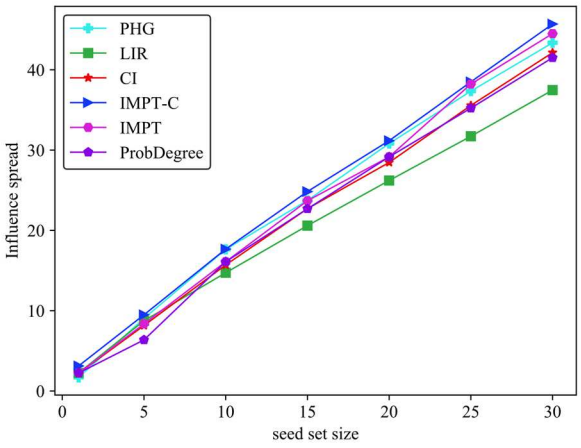
The IMPT-C algorithm is implemented and evaluated on two real-world data sets and two artificial datasets. These data sets contain graphs with different scales. The Sister cities data set contains a collection of cities in the world that are connected to each other through a "sister city" [32]. This data set is extracted from WikiData and contains 14,274 nodes and 20,573 edges. The PGP data set also includes users who communicate with each other through the Pretty Good Privacy algorithm [33]. This data set contains only one giant connected component and contains 10,680 nodes and 24,316 edges. The MFO115 artificial data set is also produced from the fire forest model with a connection probability of $p = 0.115$ and includes 10,000 nodes and 23,142 edges [34]. Also, the MFO120 artificial dataset is produced using the fire forest model with a connection probability of $p = 0.120$, including 10,000 nodes and 29566 edges [34].

B. Results

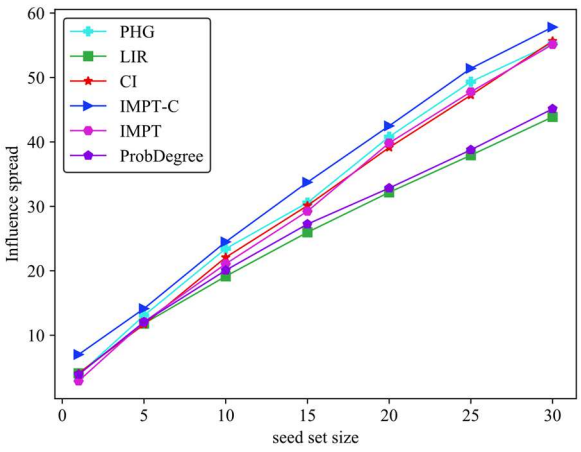
After simulation and implementation of the IMPT-C, the IS and run time of these algorithms have been compared with other state-of-the-art algorithms. Fig. 2 indicates the IS of the compared algorithms on the two data sets of Sister cities and PGP. Also, Fig. 3 shows the IS of the compared algorithms on the two artificial data sets M-FO115 and M-FO120. The number of seed set are shown on the x-axis from $k = 5$ to $k = 30$, and the IS is shown on the y-axis by k different nodes.

Fig. 2 (a) shows the IS of the different algorithms on the Sister cities data set. The IS of the IMPT-C and then the PHG is higher than the other evaluated algorithms, and these algorithms are in $k = 1$ to $k = 20$, presented results close to each other. The IMPT-C is 5.25% better than the PHG algorithm in terms of IS. Also, the IS of the IMPT is higher than the PHG algorithm in $k = 25$ to next. Then the IS in the Sister cities data set belongs to the CI algorithm, followed by the ProbDegree and LIR, respectively. Also, the results of ProbDegree and LIR are almost close to each other. According to the results of the compared algorithms on the PGP data set in Fig. 2 (b), the IS of the IMPT-C algorithm is higher than each other algorithms. Then the PHG algorithm has a higher IS. The proposed IMPT-C algorithm is 4.43% better than the PHG algorithm in terms of IS. IMPT and CI also show the IS very close to each other. The LIR also shows the lowest IS on the Sister cities data set.

Fig 3 (a) shows the IS of the compared algorithms on the M-FO115 artificial data set. In this data set, the IS of the IMPT-C is higher than other compared algorithms. The IMPT, PHG, ProbDegree, and CI have shown close results than each other; However, the IS of the LIR algorithm has significantly decreased from $k = 10$. The proposed IMPT-C algorithm is 2.36% better than the PHG algorithm in terms of IS. The IS of the algorithms on the M-FO120 artificial data set is indicated in Fig. 3 (b). In this data set, the IS of the IMPT-C algorithm is higher than other algorithms. Then the IMPT algorithm has a higher IS. Of course, the results of these two algorithms and other evaluated algorithms up to $k = 10$ are very close to each other. The proposed IMPT-C algorithm is 4.26% better than the PHG algorithm in terms of IS. The results of the IS of the LIR algorithm from $k = 5$ have significantly decreased.



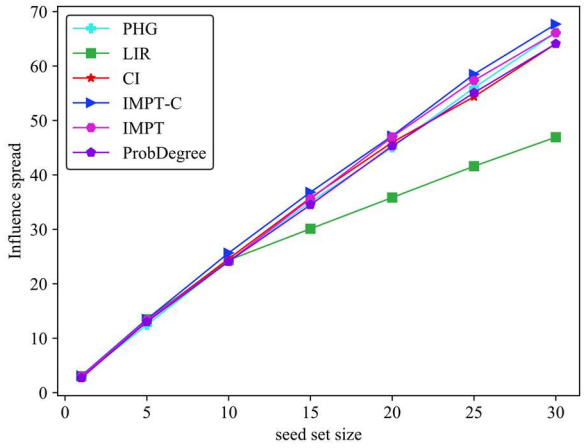
(a)



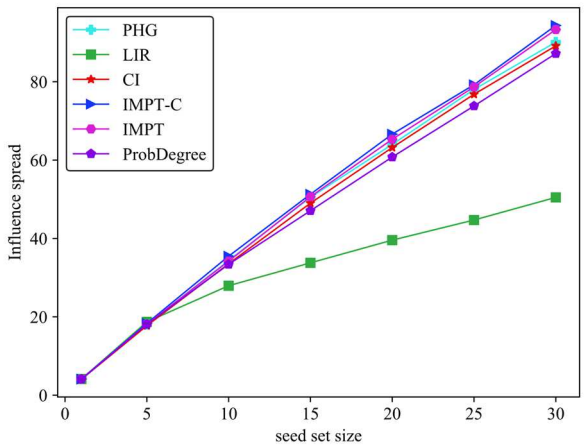
(b)

Fig. 2. The IS under IC diffusion model with $p = 0.01$ on the (a) Sister cities and (b) PGP data sets

The run time in algorithms related to IMP indicates the time to find k seed nods in set S. In Fig. 4, a diagram of the results of the run time of the proposed IMPT-C algorithm and other compared algorithms, for $k = 30$ and $R = 1000$ using the IC diffusion model, on two real-world data sets and two artificial data sets, per seconds, it has been shown. According to the run time diagram of the algorithms in Fig. 4, the lowest run time belongs to the IMPT-C algorithm. Reducing the selective space and identifying CNs in order to find the final influential nodes has caused the proposed algorithm to decrease the computational overhead. Also, the highest run time belongs to the PHG algorithm. Of course, the PHG algorithm on the Sister cities dataset shows less run time than the CI algorithm. It should be noted that the highest run time after PHG algorithm belongs to the CI algorithm on three datasets PGP, M-FO115, and M-FO120. Also, the run time of LIR and ProbDegree algorithms on two artificial data sets are almost identical.



(a)



(b)

Fig. 3. the IS under IC diffusion model with $p = 0.01$ on the (a) M-FO115 and (b) M-FO120 data sets

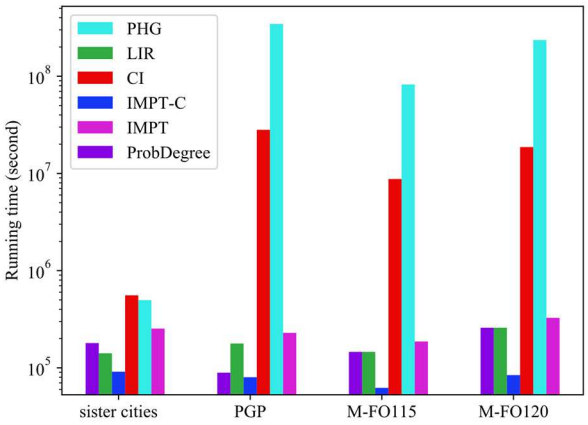


Fig. 4. The run time of different evaluated algorithms

V. CONCLUTION AND FUTURE WORKS

To solve the IMP on SNs, a small number of nodes from the graph are selected in such a way that when they are activated, the greatest influence on a huge number of people in that SN is created in the shortest time. Therefore, in this paper, the IMPT-C algorithm is presented in order to find the most influential nodes at low run time. In this algorithm, CNs are selected with a focus on graph pre-processing. This has led to a reduction in selective space and increased time efficiency. Then, based on the topology structure of the graph, influential nodes are selected. The experiment of the IMPT-C on two real-world data sets, Sister cities and PGP, on two artificial data sets, M-FO115 and M-FO120 compared to other algorithms, indicate that the IMPT-C has a high IS with low run time. As future work, the IMPT-C algorithm can be extended to make parallel processing. Also, in this algorithm, the effect of overlapping nodes on IS can be investigated.

REFERENCE

1. Liqing, Q., et al., *TSIM: A two-stage selection algorithm for influence maximization in social networks.* IEEE Access, 2020. 8: p. 12084-12095.
2. Kumar, S., et al., *IM-ELPR: Influence maximization in social networks using label propagation based community structure.* Applied Intelligence, 2021: p. 1-19.
3. Tang, J., et al., *A sequential seed scheduling heuristic based on determinate and latent margin for influence maximization problem with limited budget.* International Journal of Modern Physics C (IJMPC), 2021. 32(06): p. 1-18.
4. Beni, H.A. and A. Bouyer, *TI-SC: top-k influential nodes selection based on community detection and scoring criteria in social networks.* Journal of Ambient Intelligence and Humanized Computing, 2020: p. 1-20.
5. Aghaee, Z. and S. Kianian, *Efficient influence spread estimation for influence maximization.*

Social Network Analysis and Mining, 2020. 10(1): p. 1-21.
6. Aghaee, Z., et al., *A survey on meta-heuristic algorithms for the influence maximization problem in the social networks.* Computing, 2021: p. 1-41.
7. Chen, W., Y. Wang, and S. Yang. *Efficient influence maximization in social networks*. in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2009.
8. Kempe, D., J. Kleinberg, and É. Tardos. *Maximizing the spread of influence through a social network*. in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2003.
9. Chen, W., L.V. Lakshmanan, and C. Castillo, *Information and influence propagation in social networks.* Synthesis Lectures on Data Management, 2013. 5(4): p. 1-177.
10. Domingos, P. and M. Richardson. *Mining the network value of customers*. in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. 2001.
11. Leskovec, J., et al. *Cost-effective outbreak detection in networks*. in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2007.
12. Heidari, M., M. Asadpour, and H. Faili, *SMG: Fast scalable greedy algorithm for influence maximization in social networks.* Physica A: Statistical Mechanics and its Applications, 2015. 420: p. 124-133.
13. Liu, D., et al., *A fast and efficient algorithm for mining top-k nodes in complex networks.* Scientific reports, 2017. 7(1): p. 1-8.
14. Aghaee, Z. and S. Kianian, *Influence maximization algorithm based on reducing search space in the social networks.* SN Applied Sciences, 2020. 2(12): p. 1-14.
15. Li, W., et al., *Three-hop velocity attenuation propagation model for influence maximization in social networks.* World Wide Web, 2020. 23(2): p. 1261-1273.
16. Beni, H.A., et al. *IMT: Selection of Top-k Nodes based on the Topology Structure in Social Networks*. in *2020 6th International Conference on Web Research (ICWR)*. 2020. IEEE.
17. Aghaee, Z., et al. *A Heuristic Algorithm Focusing on the Rich-Club Phenomenon for the Influence Maximization Problem in Social Networks*. in *2020 6th International Conference on Web Research (ICWR)*. 2020. IEEE.
18. Qiu, L., et al., *PHG: A three-phase algorithm for influence maximization based on community structure.* IEEE Access, 2019. 7: p. 62511-62522.
19. Singh, S.S., et al., *C2IM: Community based context-aware influence maximization in social networks.* Physica A: Statistical Mechanics and its Applications, 2019. 514: p. 796-818.
20. Jiang, Q., et al. *Simulated annealing based influence maximization in social networks*. in *Proceedings of the AAAI Conference on Artificial Intelligence*. 2011.
21. Cui, L., et al., *DDSE: A novel evolutionary algorithm based on degree-descending search strategy for influence maximization in social networks.* Journal of Network and Computer Applications, 2018. 103: p. 119-130.
22. Gong, M., et al., *Influence maximization in social networks based on discrete particle swarm optimization.* Information Sciences, 2016. 367: p. 600-614.
23. Zareie, A., A. Sheikhahmadi, and M. Jalili, *Identification of influential users in social network using gray wolf optimization algorithm.* Expert Systems with Applications, 2020. 142: p. 112971.
24. Blondel, V.D., et al., *Fast unfolding of communities in large networks.* Journal of statistical mechanics: theory and experiment, 2008. 2008(10): p. P10008.
25. Kitsak, M., et al., *Identification of influential spreaders in complex networks.* Nature physics, 2010. 6(11): p. 888-893.
26. Page, L., et al., *The PageRank citation ranking: Bringing order to the web*. 1999, Stanford InfoLab.
27. Morone, F., et al., *Collective influence algorithm to find influencers via optimal percolation in massively large social media.* Scientific reports, 2016. 6(1): p. 1-11.
28. Shang, J., et al., *CoFIM: A community-based framework for influence maximization on large-scale networks.* Knowledge-Based Systems, 2017. 117: p. 88-100.
29. Nguyen, D.-L., et al., *Probability-based multi-hop diffusion method for influence maximization in social networks.* Wireless Personal Communications, 2017. 93(4): p. 903-916.
30. Ahmadi Beni, H. and A. Bouyer, *Identifying Influential Nodes Using a Shell-Based Ranking and Filtering Method in Social Networks.* Big Data, 2021.
31. Beni, H.A., S. Azimi, and A. Bouyer. *A node filtering approach for Influence Maximization problem in Independent Cascade model*. in *2020 6th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS)*. 2020. IEEE.
32. Kunegis, J. *Konect: the koblenz network collection*. in *Proceedings of the 22nd international conference on world wide web*. 2013.
33. Boguná, M., et al., *Models of social networks based on social distance attachment.* Physical review E, 2004. 70(5): p. 056122.
34. Barabási, A.-L. and R. Albert, *Emergence of scaling in random networks.* science, 1999. 286(5439): p. 509-512.