



Group Influence Maximization in Social Networks

Yuting Zhong¹ and Longkun Guo^{1,2}(✉) 

¹ College of Mathematics and Computer Science, Fuzhou University, Fuzhou, China
longkun.guo@gmail.com

² School of Computer Science, Qilu University of Technology, Jinan, China

Abstract. In emerging applications of social networks, groups play a vital role as most decisions are made by groups according to the opinion of the majority therein. This brings the problem of Group Influence Maximization (GIM) which aims to select k initial active nodes for maximizing the expected number of influenced groups. In the paper, we study GIM and focus on activating groups rather than individuals. Observing the known NP-hardness of GIM and the $\#P$ -hardness of computing the objective function under Independent Cascade (IC) model, we devise an algorithm called Complementary Maximum Coverage (CMC) based on analyzing the influence of the nodes over the groups, ensuring the task of maximizing the number of activated groups. In addition, we also propose an algorithm called Improved Reverse Influence Sampling (IRIS) via adjusting the famous Reverse Influence Sampling (RIS) algorithm for GIM. Lastly, experiments are carried out to demonstrate that our CMC and IRIS both outperform the known baselines including Maximum Coverage and Maximum Out-degree algorithms in the average number of activated groups under IC model.

Keywords: Complementary Maximum Coverage (CMC) algorithm · Improved Reverse Influence Sampling (IRIS) algorithm · Group Influence Maximization (GIM) · Independent Cascade (IC) model

1 Introduction

With the development of the Internet and the continuous improvement of information technology, people are more and more inclined to conduct social activities on the Internet, and express their views or share their daily life on social software, thus giving rise to social networks mediated by social software. As a platform, social networks play an important role in the interaction between individuals and the dissemination of information and ideas. Among the commonly used social software, Facebook has 2.2 billion users, WeChat has 1 billion users, and Twitter has 340 million users [1]. People are not lonely in the real life as well as in social networks. They will form groups based on their common interests and hobbies or some kind of relationship. The group can be large or small: as

small as two or three people, such as a family; or as large as a state or even a country. Because social networks have such important influence, they have great applications in information dissemination, advertising marketing, public opinion control and other aspects. We take viral marketing as an example, when advertising companies want to achieve a good marketing result with limited cost, they will use the “word-of-mouth” effect to select k users to maximize advertising audiences through their “mouths”. In addition, individual decisions depend on group decisions. For example, a company needs to buy pens of a certain brand for all employees. When a majority of employees are promoted by a certain brand and decide to buy pens of that brand, the company will buy all pens of that brand. Another example is the US presidential election. If a presidential candidate wins a majority in a state election, he or she will have all the electoral votes of the state. This is also a practical example of maximizing group influence.

1.1 Related Work

To the best of our knowledge, Domingos and Richardson et al. [2] were the first to address the issue of Influence Maximization (IM). Kempe et al. [3] were the first to formulate IM as a discrete optimization problem. They showed the IM problem is NP-hard under either the Independent Cascade (IC) model or Linear Threshold (LT) model, while the objective function is submodular. They proposed a greedy algorithm using Monte Carlo method to simulate the process of influence propagation, which achieved $(1 - \frac{1}{e} - \varepsilon)$ approximation solution for any $\varepsilon > 0$. However, the computation time of the greedy algorithm is expensive. Subsequently, more and more scholars have dedicated themselves to the study of IM problem and proposing approximation algorithms based on improved greedy approach and heuristic algorithms [4–8]. Heuristic algorithms are favored by scholars because of its fast computation speed, but its approximate quality is not as good as the approximation algorithms. In particular, the improved greedy algorithms are faster than the traditional greedy algorithms via making use of the submodularity of the objective function.

Due to the further study of many scholars, the algorithms for solving IM problem have also developed rapidly. Among them, Borgs et al. [9] proposed RIS algorithm, which greatly reduced the computational time of the simulation propagation process. Based on RIS algorithm, Tang et al. [10, 11] proposed TIM, TIM⁺ and IMM algorithms, which guaranteed a $(1 - \frac{1}{e} - \varepsilon)$ approximation ratio under IC model. Recently, Nguyen et al. [12] proposed the SSA and D-SSA algorithms, which were the first approximation algorithm which satisfies the strict theoretical threshold of IM with the minimum sample set.

A social network is divided into multiple communities by community discovery algorithms. The nodes within communities are closely connected while the nodes among communities are sparsely connected, so the influence within communities spreads quickly in a wide range. The two most commonly used community discovery algorithms are OASNET (Optimal Allocation in a Social NETWORK) algorithm and CGA (Community-based Greedy Algorithm) algorithm.

The OASNET algorithm was proposed by Cao et al. [13] to solve the IM problem by using the optimal dynamic allocation of resources. CGA algorithm [14] combined the dynamic programming method and the greedy algorithm to allocate the optimal number of seed nodes for each community so as to maximize the influence. Ji et al. [15] proposed a new algorithm, which found out the hidden community structure in the network and then selected k nodes with the largest number of community coverage as seed nodes. Moreover, many researchers are committed to studying the property of community to solve IM aiming to maximize the number of eventually activated nodes, while the task of GIM is to activate maximum groups rather than individuals [16–22].

In terms of research on GIM, Zhu et al. [23] proposed a sandwich approximation framework based on D-SSA method to obtain seed nodes, which achieved an approximation guarantee of $(1 - \frac{1}{e} - \varepsilon)$. In addition, Zhu et al. [24] also proposed a sandwich approximation framework based on ED-SSA method, which approximated the upper and lower bounds of the objective function and compared them with the Group Coverage Maximization Algorithm (GCMA) to obtain the seed nodes. Although great breakthroughs have been made in GIM, many scholars are still working on more efficient algorithms.

1.2 Our Contribution

Our results can be summarized as in the following:

- We devise a heuristic algorithm called Complementary Maximum Coverage (CMC), which emphasizes the influence of the nodes over groups to solve GIM.
- We also propose the Improved Reverse Influence Sampling (IRIS) algorithm by adjusting the famous Reverse Influence Sampling (RIS) algorithm for GIM.
- Compared with Maximum Coverage (MC) algorithm and Maximum Out-degree (MO) algorithm by experiments, our proposed algorithms outperform both MC and MO regarding the average number of eventually activated groups under the IC model.

1.3 Organization

The remainder of the paper is organized as below: Sect. 2 gives the social network model, and formally introduces the GIM problem; Sect. 3 presents the CMC algorithm and IRIS algorithm; Sect. 4 evaluates the four algorithms under IC model through numerical experiments; Sect. 5 concludes the paper.

2 Problem Description

2.1 Network Model

We model the social network as $G = (V, E, P, U)$.

V represents the set of nodes which represent users in a social network. Assume that the social network has n users, then $V = \{v_1, v_2, \dots, v_n\}$. The node can have influence on other nodes or be influenced by other nodes, which form edges. If there's an edge between two nodes, we could say that the two nodes are neighbors to each other.

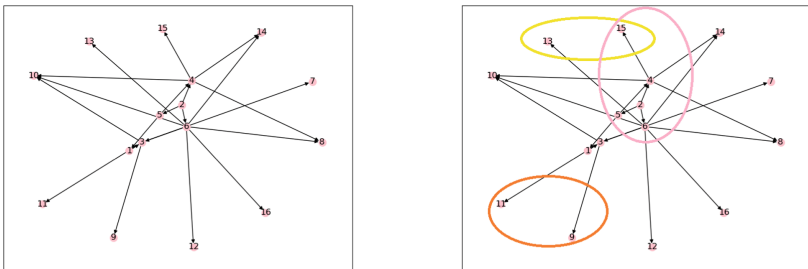
E represents the set of edges which represent the influence between nodes. Assume that the social network has m edges, then $E = \{e_1, e_2, \dots, e_m\}$. The edge can be directed or undirected. For example, in the directed graph, (u, v) means that node u has influence on node v , but node v has no influence on node u , u is the source node and v is the target node. The edge taking u as the source node is u 's outgoing edge, and the edge taking u as the target node is u 's entry edge. The sum of the number of u 's outgoing edges is the out-degree of u , and the sum of the number of u 's entry edges is the in-degree of u .

P represents the set of probabilities which are the weights of the edges, then $P = \{p_1, p_2, \dots, p_m\}$, and $\forall p_i \in [0, 1], 1 \leq i \leq m, i$ is a positive integer. The higher the probability is, the more likely the source node is to successfully activate the target node.

U represents the set of groups. Assume that the social network has l groups, then $U = \{u_1, u_2, \dots, u_l\}$, and u_j is a subset of $V, 1 \leq j \leq l, j$ is a positive integer. In a social network, each node can be an individual or belong to one or more groups. When $\beta\%$ of the members in a group are affected, we assume that the group is successfully affected.

2.2 Group Influence Maximization

The IM problem is to study the maximum number of nodes that will be activated with k initial active nodes under the given information diffusion model. Figure 1a is a simple social network graph without groups. Each edge is directed, indicating that the influence flows from the source node to the target node, and each edge is probabilistic.



(a) A simple social network without groups (b) A simple social network with groups

Fig. 1. Examples of simple social networks with vs without groups (Color figure online)

The GIM aims to seek k nodes to maximize the expected number of eventually activated groups. Each node in GIM can be independent or belong to one or more groups, and a group will be activated only if the $\beta\%$ members of the group are activated. The larger the value of β is, the more difficult it is for the group to be activated.

The IM problem is a special example of the GIM problem. Each group in GIM represents each node in IM, and $\beta\% = 100\%$. The IM problem is NP-hard. Obviously, the GIM problem is also NP-hard. For a given graph $G = (V, E, P, U)$, the mathematical description of GIM is:

$$\begin{aligned} \max \quad & \rho(S) \\ \text{s.t.} \quad & |S| \leq k \end{aligned}$$

where S is the set of seed nodes, k is the number of initial seed nodes, and $\rho(S)$ is the expected number of groups activated by initial seed nodes under a given propagation model. It is difficult to calculate $\rho(S)$ because the activation is probabilistic and random. In the IM problem, computing $\rho(S)$ under IC model is #P-hard, likewise, in the GIM problem, computing $\rho(S)$ under IC model is also #P-hard [24].

Obviously, activating more nodes is not the same as activating more groups. As shown in Fig. 1b, there are three groups in the social network, that is $U = \{u_1, u_2, u_3\}$, u_1 is the yellow one, u_2 is the pink one, u_3 is the orange one. $u_1 = \{13, 15\}$, $u_2 = \{2, 4, 5, 6, 15\}$, $u_3 = \{9, 11\}$. The group activation threshold is assumed to be 50%, meaning that the group will be activated only when at least half of members of the group are active. For example, it is assumed that the seed node $\{2\}$ finally activates $\{2, 4, 15\}$, three nodes are activated, so u_1 and u_2 are activated, $\rho(S) = 2$. Another case is that seed node $\{2\}$ successfully influences $\{2, 4, 5, 6\}$, four nodes are activated, but only u_2 is activated, $\rho(S) = 1$. Therefore, activating more nodes is not the same as activating more groups. But the more nodes are activated, the more likely groups are to be activated.

3 The Algorithms for Solving GIM

In this section we discuss the algorithms to solve Group Influence Maximization (GIM) in the paper, including Complementary Maximum Coverage (CMC) algorithm and Improved Reverse Influence Sampling (IRIS) algorithm.

3.1 Complementary Maximum Coverage Algorithm

The CMC algorithm is the complementary of MC algorithm. MC algorithm aims to seek k seed nodes with maximum group coverage. However, MC algorithm does not take the contribution of nodes over groups into account. If a node covers maximum groups, but those groups require $\beta\%$ active members to be activated, and this node is only a member of the large groups. In case the node does not activate other members of the groups, then the node makes little contribution to

Algorithm 1. The Complementary Maximum Coverage algorithm

Input: An instance of GIM $G = (V, E, P, U)$, the number of seed nodes k , the activation threshold of groups β .

Output: the seed set S .

- 1: Set $S := \emptyset$;
- 2: Calculate the f_c of each node following Equality 1.
- 3: Sort the nodes according to the computed f_c .
- 4: Add k nodes with maximum f_c into S .
- 5: **return** S .

the groups. The idea of CMC algorithm is to treat all the nodes as seed nodes, then remove $n - k$ seed nodes with the least influence over groups, and finally obtain k seed nodes. The influence of a node on a group is not only reflected in whether deleting the node has an impact on activating the group, but also in whether it can activate other members in the group. We use $f_c(v_i)$ to calculate the influence of v_i over groups which it covers. If a node does not belong to any group, its $f_c = 0$. If a node covers more than one group, then f_c equals the sum of its influence on groups. If the node has maximum group coverage, the f_c of the node may be larger. We have $f_c(v_i)$

$$f_c(v_i) = \sum_j \frac{a_{v_i}}{|u_j| - H_{u_j} + 1} \quad (1)$$

where j is the group number, u_j is the group covered by v_i , a_{v_i} is the number of members that v_i activates successfully in u_j , including v_i itself. All the nodes activated by v_i are obtained by breadth first search (BFS) method, then calculate the number of these nodes in u_j , the result is a_{v_i} . a_{v_i} measures the active degree of v_i in the group. The larger a_{v_i} is, the more members of the group can be activated, which also increases the possibility of activating the group. $|u_j|$ is the total number of members of u_j , H_{u_j} is the activation threshold of u_j , and $H_{u_j} = \beta\% \times |u_j|$. $|u_j| - H_{u_j}$ means that u_j allows $|u_j| - H_{u_j}$ nodes to be deleted, and the larger $|u_j| - H_{u_j}$ is, the less influence v_i has on u_j . Due to the denominator can't be zero, we define the denominator to be $|u_j| - H_{u_j} + 1$.

Lemma 1 *The runtime of Algorithm 1 is $O(nl + n + m)$.*

CMC performs the following operations on the nodes numbered from 1 to n : traverse l groups and find out the groups covered by each node. The runtime of the step is $O(nl)$. Then compute the number of activated members within the groups covered by each node via BFS method, so the runtime is $O(n + m)$. Hence, the runtime of the CMC algorithm is $O(nl + n + m)$.

3.2 Improved Reverse Influence Sampling Algorithm

The Improved Reverse Influence Sampling (IRIS) algorithm is improved on the basis of Reverse Influence Sampling (RIS) algorithm. The RIS algorithm

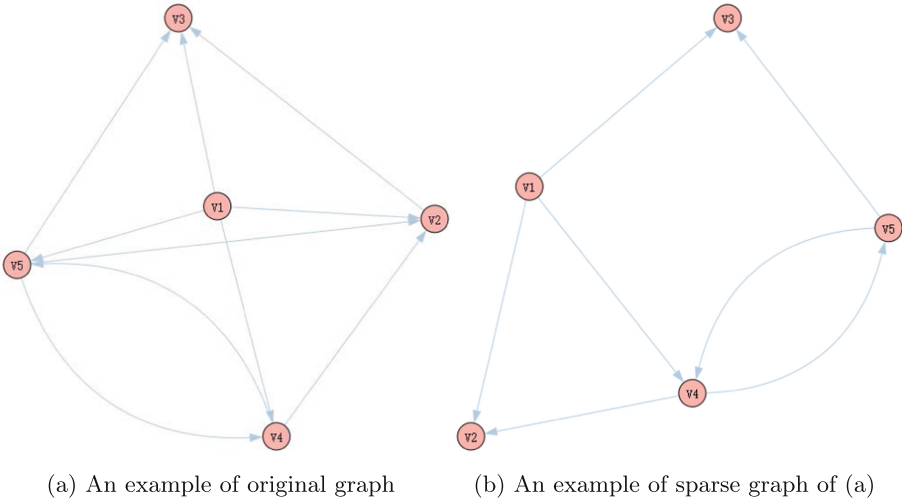


Fig. 2. An example of a RR set generation

Algorithm 2. get_RRS algorithm

Input: An instance of GIM $G = (V, E, P, U)$.

Output: RR_set.

- 1: RR_set, new_nodes := \emptyset .
 - 2: Get a random graph g from G .
 - 3: Choose node v from g uniformly at random.
 - 4: Add v into new_nodes.
 - 5: **repeat**
 - 6: Simulate influence spread, starting from new_nodes.
 - 7: Add source nodes of new_nodes into RR_set.
 - 8: Update new_nodes.
 - 9: **until** new_nodes is empty.
 - 10: **return** RR_set.
-

is divided into two steps: the process of generating RR (Reverse Reachable) sets and the process of selecting seed nodes. The first step is to randomly select node v in the original graph and traverse the entry edge of v . Each edge is inverted with the probability of p , or remains unchanged with the probability of $1-p$. Finally, a sparse reverse graph is generated. This helps to keep the high-probability edges, allowing a wider range of propagation. Simply speaking, the set of nodes that can reach node v with high probability is the RR set of node v . To take a simple example, Fig. 2a is the original social network graph, and there are 5 nodes and 10 directed edges. Figure 2b is the sparse graph of Fig. 2a, leaving 7 edges with high probability. The RR set of node v_2 is $\{v_2, v_1, v_4, v_5\}$, where each node has high probability to activate v_2 .

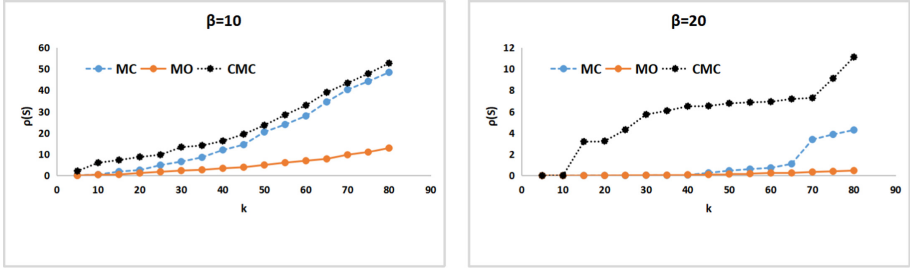


Fig. 3. Comparison of CMC, MC and MO under IC model for Dataset1

The second step of the RIS algorithm is to select the seed nodes covering maximum RR sets. Because covering more RR sets means affecting more nodes. The k nodes that cover most RR sets are the seed nodes we are looking for. But now the task is to activate maximum groups, not most nodes, and activating more nodes does not mean activating more groups. So we propose IRIS in order to solve GIM, we change the second step of RIS to choose k nodes that have maximum group coverage. In this way, the selected seed nodes can not only have certain propagation influence to activate more nodes, but also can activate more groups.

Lemma 2 *The runtime of Algorithm 3 is $O(\Gamma(n+m) + knl)$.*

The IRIS algorithm first forms Γ random sparse graphs of G , then randomly selects a node to generate a RR set in each subgraph. The runtime of the first step is $O(\Gamma(n+m))$. Secondly iterate k times to select the node that covers the most groups in the RR sets. The runtime of the second step is $O(knl)$. Thus the runtime of the IRIS algorithm is $O(\Gamma(n+m) + knl)$.

Algorithm 3. The Improved Reverse Influence Sampling algorithm

Input: An instance of GIM $G = (V, E, P, U)$, the number of seed nodes k , Monte Carlo times t .

Output: the seed set S .

- 1: Set $S := \emptyset$;
 - 2: $R \leftarrow$ generate t RR sets by Algorithm 2
 - 3: **for** $i = 1$ to k **do**
 - 4: Add the node that has the maximum group coverage in R into S .
 - 5: Delete RR sets that contain S in R .
 - 6: **end for**
 - 7: **return** S .
-

4 Numerical Experiments

4.1 Experimental Setting

We used two data sets to perform experiments under the Independent Cascade (IC) model, including the undirected graph Dataset1 and directed graph Dataset2. Dataset1 collected in March 2020 is a social network of users from Asian (e.g. Philippines, Malaysia, Singapore) countries [25]. Nodes represent users of the music streaming service LastFM and links among them are friendships. Dataset2 consists of 9 snapshots of the Gnutella peer-to-peer file sharing network from August 2002 from SNAP. Nodes represent hosts in the Gnutella network topology and edges represent connections between the Gnutella hosts. For the convenience of the experiments, groups were randomly generated, and the probability of each edge was randomly generated. The Improved Reverse Influence Sampling (IRIS) algorithm is applied to directed graphs, so Dataset2 is available for the four algorithms, while Dataset1 is suitable to other algorithms except IRIS. Table 1 is the information of data sets used in our experiments.

Table 1. Datasets information

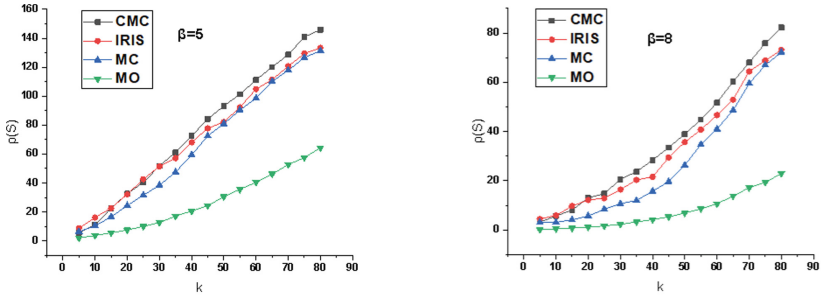
	Type	Nodes	Edges	Groups	Average group size
Dataset1	Undirected	7624	27806	198	34.01
Dataset2	Directed	6301	20777	234	33.84

Because k and β affect the objective function. Therefore we set the value of k from 5 to 80 at an interval of 5 for Dataset1 and Dataset2. Set the value of β to 10 and 20 for Dataset1, set the value of β to 5, 8, 10, 12, 15, 18 respectively for Dataset2. All programs were written in python3.7.

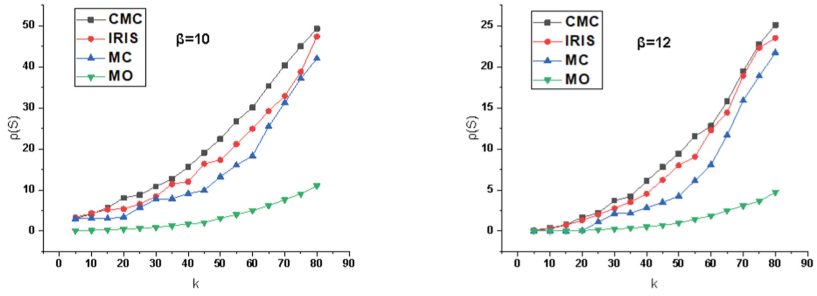
4.2 Experimental Results

As can be seen from Fig. 3, when the size of k grows, the number of activated nodes increases and hence does the number of activated groups. Besides, the number of groups activated decreases when β grows. For the undirected graph Dataset1, CMC outperforms MC and MO in general. MO performs worst, because MO aiming to seek k nodes with maximum out-degree does not focus on group activation. The performances of CMC and MC have little difference when $\beta = 10$, because the activation threshold is small, the two algorithms can find the key nodes to activate the majority of the groups. When $\beta = 20$, the difference between the experimental results of the two algorithms is widened, because the activation threshold increases, group activation becomes difficult, and the shortcomings of MC algorithm are also revealed.

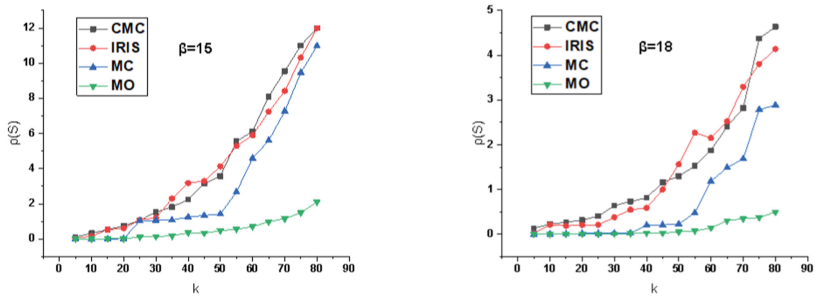
As for the directed graph Dataset2, we can find that CMC and IRIS have better performance than MC and MO in average. From Fig. 4a CMC performs



(a) $\beta = 5$ and $\beta = 8$



(b) $\beta = 10$ and $\beta = 12$



(c) $\beta = 15$ and $\beta = 18$

Fig. 4. Comparison of CMC, IRIS, MC and MO under IC model for Dataset2

best, IRIS is closer to MC, but slightly better than MC. While in Fig. 4b, the gap between IRIS and MC is widening, and IRIS is closer to CMC. Because IRIS not only focuses on seeking nodes which can cover more groups, but also attaches importance to propagation influence.

As demonstrated in Fig. 4c, our CMC algorithm has the best performance among all the algorithms in almost every instance except for some rare ones. The exceptional cases happen mainly because seed nodes obtained by IRIS differ in each time and the Reverse Reachable (RR) sets computed by IRIS focus on the influence of nodes outside the group, while CMC emphasizes the influence of nodes on members within the group.

5 Conclusion

In this paper, we proposed a heuristic algorithm called Complementary Maximum Coverage (CMC) based on analyzing the influence of the nodes over the groups to ensure the task of maximizing the number of groups activated. In addition, we also presented an algorithm called Improved Reverse Influence Sampling (IRIS) which is derived via improving the famous algorithm called Reverse Influence Sampling (RIS). Through experiments, we demonstrated that both CMC and IRIS outperform Maximum Coverage (MC) and Maximum Out-degree (MO) algorithms regarding the average number of activated groups under Independent Cascade model. Further, the CMC algorithm has better performance than IRIS in most cases besides the case when $\beta \geq 15$, while it runs significantly faster than IRIS in all instances. This indicates that CMC is the best among all the four algorithms. However, the deficiencies of CMC algorithm are that the effect is not significant when β is low and CMC is the third fast that it is slightly slower than MO and MC. We are currently analyzing the theoretical performance of CMC so as to provide an approximation ratio for the algorithm.

Acknowledgement. The authors are supported by National Natural Science Foundation of China (No.61772005) and Natural Science Foundation of Fujian Province (No.2017J01753).

References

1. Mary, M., Wu, L.: Internet trends 2018 (2018)
2. Domingos, P., Richardson, M.: Mining the network value of customers, pp. 57–66 (2001)
3. Kempe, D., Kleinberg, J., Tardos, E.: Maximizing the spread of influence through a social network, pp. 137–146 (2003)
4. Leskovec, J., Krause, A., Guestrin, C., Faloutsos, C., Vanbriesen, J.M., Glance, N.: Cost-effective outbreak detection in networks, pp. 420–429 (2007)
5. Goyal, A., Lu, W., Lakshmanan, L.V.S.: Celf++: optimizing the greedy algorithm for influence maximization in social networks, pp. 47–48 (2011)
6. Chen, W., Wang, Y., Yang, S.: Efficient influence maximization in social networks, pp. 199–208 (2009)

7. Wasserman, S., Faust, K.: *Social Network Analysis: Social Network Analysis in the Social and Behavioral Sciences*. Cambridge University Press, Cambridge (1994)
8. Estevez, P.A., Vera, P.A., Saito, K.: Selecting the most influential nodes in social networks, pp. 2397–2402 (2007)
9. Borgs, C., Brautbar, M., Chayes, J., Lucier, B.: Maximizing social influence in nearly optimal time, pp. 946–957 (2014)
10. Tang, Y., Xiao, X., Shi, Y.: Influence maximization: near-optimal time complexity meets practical efficiency, pp. 75–86 (2014)
11. Tang, Y., Shi, Y., Xiao, X.: Influence maximization in near-linear time: a martingale approach, pp. 1539–1554 (2015)
12. Nguyen, H.T., Thai, M.T., Dinh, T.N.: Stop-and-stare: optimal sampling algorithms for viral marketing in billion-scale networks, pp. 695–710 (2016)
13. Cao, T., Wu, X., Wang, S., Hu, X.: Oasnet: an optimal allocation approach to influence maximization in modular social networks, pp. 1088–1094 (2010)
14. Wang, Y., Cong, G., Song, G., Xie, K.: Community-based greedy algorithm for mining top-k influential nodes in mobile social networks, pp. 1039–1048 (2010)
15. Ji, J., Huang, L., Wang, Z., Li, H., Li, S.: A new approach to maximizing the spread of influence based on community structure. *J. Jilin Univ. (Science Edition)* **1**, 23 (2011)
16. Shuang, W.A.N.G., Bin, L.I., Xuejun, L.I.U., Ping, H.U.: Division of community-based influence maximization algorithm. *Computer Engineering and Applications* **2016**(19), 8 (2016)
17. Shang, J., Zhou, S., Li, X., Liu, L., Hongchun, W.: Cofim: a community-based framework for influence maximization on large-scale networks. *Knowl. Based Syst.* **117**, 88–100 (2017)
18. Xie, J., Szymanski, B.K., Liu, X.: SLPA: uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process. In: 2011 IEEE 11th International Conference on Data Mining Workshops, pp. 344–349. IEEE (2011)
19. Qiu, L., Jia, W., Fan, X.: Influence maximization algorithm based on overlapping community. *Data Anal. Knowl. Disc.* **3**(7), 94–102 (2019)
20. Chen, Y.-C., Zhu, W.-Y., Peng, W.-C., Lee, W.-C., Lee, S.-Y.: CIM: community-based influence maximization in social networks. *ACM Trans. Intell. Syst. Technol. (TIST)* **5**(2), 1–31 (2014)
21. Huang, H., Shen, H., Meng, Z., Chang, H., He, H.: Community-based influence maximization for viral marketing. *Appl. Intell.* **49**(6), 2137–2150 (2019). <https://doi.org/10.1007/s10489-018-1387-8>
22. Bozorgi, A., Samet, S., Kwisthout, J., Wareham, T.: Community-based influence maximization in social networks under a competitive linear threshold model. *Knowl. Based Syst.* **134**, 149–158 (2017)
23. Zhu, J., Zhu, J., Ghosh, S., Weili, W., Yuan, J.: Social influence maximization in hypergraph in social networks. *IEEE Trans. Netw. Sci. Eng.* **6**(4), 801–811 (2018)
24. Zhu, J., Ghosh, S., Weili, W.: Group influence maximization problem in social networks. *IEEE Trans. Comput. Soc. Syst.* **6**(6), 1156–1164 (2019)
25. Rozemberczki, B., Sarkar, R.: Characteristic functions on graphs: birds of a feather, from statistical descriptors to parametric models (2020). arXiv preprint [arXiv:2005.07959](https://arxiv.org/abs/2005.07959)